Multifocus Image Fusion using Wavelet Transform, Gradient and Mathematical Morphology

By
Muhammad Iqbal
[2006-NUST-MS PhD-CSE (E)-01]

Submitted to the Department of Computer Engineering
in partial fulfillment of the requirements for the degree of

Master of Science
in
Computer Software Engineering

Advisor
Brig. Dr Muhammad Younus Javed

College of Electrical & Mechanical Engineering
National University of Sciences and Technology
2008

بسم الله الرحمن الرحيم

In the name of ALLAH, the most Beneficent, the most Merciful.

ABSTRACT

Image fusion has its applications in many fields such as computer vision, automatic object detection, robotics, remote sensing, military and law enforcement, medical imaging and manufacturing. The objective of image fusion is to generate a resultant fused image from a set of input images (of the same scene) which describes the scene better than any single input image with respect to some relevant properties. The fused image is obtained by extracting all the useful information from the source images while not introducing artifacts or inconsistencies which will distract human observers or the following processing. For this purpose a new image fusion technique that is actually integration of multi-scale wavelet transform, gradient and mathematical morphology schemes, has been proposed. The proposed scheme's implementation mainly consists of five steps. The first step is the application of discrete wavelet transform on the set of multifocused source images. The second step deals with the computation of local gradient of each detailed wavelet coefficient block. Finding of image activity level is the next step. Generation of binary decision map takes place based on image activity levels obtained at the previous step. Different morphological operations have been performed on binary decision map that separate the focus and defocused parts of the input images. Finally, fused image has been achieved by using the processed binary decision map.

The empirical results on standard test images (i.e. Lena, Barbara, Gold Hill and Peppers) provide higher Peak Signal to Noise Ratio (PSNR) and smaller Root Mean Square Error (RMSE) values than some of the previous approaches. These fusion results strengthen the idea of using combination of multi-scale wavelet transform, gradient and mathematical morphology schemes for multifocus image fusion.

MATLAB 7.0 has been used for the implementation of the proposed approach. Experiments have been carried out on a variety of standard greyscale images with different defocus parts.

Dedicated to

My Parents and Teachers

ACKNOWLEDGEMENTS

I am thankful to Allah Almighty for giving me the ability to finish this research work and allow me to acquire knowledge from it.

I am thankful to my supervisor Brig. Dr. Muhammad Younus Javed for his guidance and help and all the committee members for their time and diligence.

Finally, I want to thank my parents and my brother for their continuous encouragement, understanding and moral support; and have a deep gratitude to Muhammad Azhar Iqbal for his continuous guidance during this research work.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

1.1 Image Fusion

Image fusion has its applications in many fields such as computer vision, automatic object detection, robotics, remote sensing, military and law enforcement, medical imaging and manufacturing. Image fusion can be defined as the combination of a set of input images of the same scene, from different sources, with the aim to obtain new or more precise knowledge about the scene, which is more suitable for human and machine perception or further image-processing tasks such as segmentation, feature extraction and object recognition [1]. The objective of image fusion is to generate a resultant fused image which describes the scene better than any single input image with respect to some relevant properties, by extracting all the useful information from the source images, and not introducing artifacts or inconsistencies which will distract human observers or the following processing.

The benefits of image fusion include:

i. Extended range of operations: multiple sensors that operate under different operating conditions can be deployed to extend the effective range of operations. For example different sensors can be used for day/night operation.
ii. Extended spatial and temporal coverage: joint information from sensors that differ in spatial resolution can increase the spatial coverage. The same is true for the temporal dimension.
iii. Reduced uncertainty: joint information from multiple sensors can reduce the uncertainty associated with the sensing or decision process.
iv. Increased reliability: the fusion of multiple measurements can reduce noise and therefore improve the reliability of the measured quantity.

v. Robust system performance: redundancy in multiple measurements can help in the system's robustness. In case one or more sensors fail or the performance of a particular sensor deteriorates, the system can depend on the other sensors.

vi. Compact representation of information: fusion leads to compact representations. For example, in remote sensing, instead of storing imagery from several spectral bands, it is comparatively more efficient to store the fused information.

1.2 Abstraction Levels in Image Fusion

Image fusion processes are often categorized as low, intermediate or high level fusion depending on the processing stage at which fusion takes place. Low level fusion, also called pixel level fusion, combines several sources of raw data (pixels) to produce new raw data that is expected to be more informative and synthetic than the inputs. In intermediate level fusion or feature level fusion, various features such as edges, corners, lines, texture parameters, etc. are combined into a feature map that may then be used by further processing. High level fusion or decision level fusion combines the results from multiple algorithms to yield a final fused decision.

1.2.1 Pixel Level Image Fusion

Pixel level fusion combines several sources of raw data to produce new raw data that is expected to be more informative and synthetic than the inputs [2-4]. Typically, in image processing, images presenting several spectral bands of the same scene are fused to produce a new image that ideally contains in a single channel all/(most) of the information available in the various spectral bands. An operator (or an image processing algorithm) could then use this single image instead of the original images. This is particularly important when the number of available spectral bands becomes so large that it is impossible to look at the images separately. This kind of fusion requires a precise (pixel-level) registration of the available images. This registration is intrinsic when the various bands come from the same sensor but it is a lot more complicated when several different sensors are used (SAR, IR scanner, camera, etc.).

1.2.2 Feature Level Image Fusion

Feature level fusion combines various features. These features may come from several raw data sources (several sensors, different moments, etc.) or from the same raw data [5-8]. In the latter case, the objective is to find relevant features among available features that might come from several feature extraction methods. The objective is to obtain a limited number of relevant features. Typically, in image processing, feature maps are computed as pre-processing for segmentation or detection. Features such as edges, corners, lines, texture parameters (Haralick, Wavelet coefficients, etc.) are computed and combined in a fused feature map that may then be used for segmentation or detection.

1.2.3 Decision Level Image Fusion

Decision level fusion combines decisions coming from several algorithms and experts. Methods of decision levels fusion include voting methods, statistical methods, fuzzy logic based methods, etc.

The above categorization does not encompass all possible fusion paradigms, as input and output of the fusion process may present different levels of processing. Typically features could be fused to output a decision. In practical problems, the applied fusion procedure is often a combination of the previously mentioned three levels.

1.3 Image Registration

Very often, there are some issues that have to be dealt with before the fusion can be performed. Most of the time, the input images are misaligned. Image registration [9-13] is used to establish a spatial correspondence between the input images and to determine a spatial geometric transformation, called warping, which aligns the images.

Misalignment of image features is caused by several factors including the geometries of the sensors, different spatial positions of the sensors, different temporal capture rates of the sensors and the inherent misalignment of the sensing elements. Image registration techniques align the images by exploiting the similarities between the input images.

1.4 Image Fusion Methods

The most important issue concerning image fusion is to determine how to combine the input images. In recent years, several image fusion techniques have been proposed. The primitive fusion schemes perform the fusion right on the source images. One of the simplest of these image fusion methods just takes the pixel-by-pixel gray level average of the source images. This simplistic approach often has serious side effects such as reducing the contrast. Some of the prominent image fusion methods are discussed in the following subsections.

1.4.1 Linear Superposition

Probably, the most straightforward way to build a fused image of several input images is performing the fusion as a weighted superposition of all input images.

The optimal weighting coefficients, with respect to information content and redundancy removal, can be determined by a principal component analysis (PCA) of all input intensities. By performing a PCA of the covariance matrix of input intensities, the weightings for each input frame are obtained from the eigenvector corresponding to the largest eigen value.

1.4.2 Non-Linear Methods

Another simple approach to image fusion is to build the fused image by the application of a simple nonlinear operator such as max or min. If in all input images the bright objects are of interest, a good choice is to compute the fused image by a pixel-by-pixel application of the maximum operator.

An extension to this approach follows by the introduction of morphological operators such as opening or closing. One application is the use of conditional morphological operators by the definition of highly reliable 'core' features present in both images and a set of 'potential' features present only in one source, where the actual fusion process is performed by the application of conditional erosion and dilation operators.

A further extension to this approach is image algebra, which is a high-level algebraic extension of image morphology, designed to describe all image processing operations. The basic types defined in image algebra are value sets, coordinate sets which allow the integration of different resolutions and tessellations, images and templates. For each basic type, binary and unary operations are defined which reach from the basic set operations to more complex ones for the operations on images and templates. Image algebra has been used in a generic way to combine multisensor images.

1.4.3 Artificial Neural Networks

Inspired by the fusion of different sensor signals in biological systems, many researchers have employed artificial neural networks in the process of image fusion [14-17]. The most popular example for the fusion of different imaging sensors in biological systems is described by Newman and Hartline in the 80s: Rattlesnakes (and the general family of pit vipers) possess so called pit organs which are sensitive to thermal radiation through a dense network of nerve fibers. The output of these pit organs is fed to the optical tectum, where it is combined with the nerve signals obtained from the eyes. Newman and Hartline distinguished six different types of bimodal neurons merging the two signals based on a sophisticated combination of suppression and enhancement.

Several researchers modeled this fusion process for the combination of multi spectral imagery by a combination of several neural networks.

1.4.4 Image Pyramids

A generic image pyramid is a sequence of images where each image is constructed by low pass filtering and subsampling from its predecessor [18-19]. Due to sampling, the image size is halved in both spatial directions at each level of the decomposition process, thus leading to a multiresolution signal representation. The difference between the input image and the filtered image is necessary to allow an exact reconstruction from the pyramidal representation. The image pyramid approach thus leads to a signal representation with two pyramids: the smoothing pyramid containing the averaged pixel values, and the difference pyramid containing the pixel differences, i.e. the edges. So the difference pyramid can be viewed as a multiresolution edge representation of the input image. The actual fusion process can be described by a generic multiresolution fusion scheme which is applicable both to image pyramids and the wavelet approach.

There are several modifications of this generic pyramid construction method described above. Some authors propose the computation of nonlinear pyramids, such as the ratio and contrast pyramid, where the multiscale edge representation is computed by a pixel-by-pixel division of neighboring resolutions. A further modification is to substitute the linear filters by morphological nonlinear filters, resulting in the morphological pyramid. Another type of image pyramid - the gradient pyramid - results, if the input image is decomposed into its directional edge representations using directional derivative filters.

1.4.5 Wavelet Transform

An image analysis method similar to image pyramid is the discrete wavelet transform [20-23]. The main difference is that while image pyramids lead to an over complete set of transform coefficients, the wavelet transform results in a non-redundant image representation.

The discrete two dimensional wavelet transform is computed by the recursive application of low pass and high pass filters in each direction of the input image (i.e. rows and columns) followed by subsampling.

The actual fusion process can be described by a generic multiresolution fusion scheme which is applicable both to image pyramids and the wavelet approach.

1.4.6 Generic Multiresolution Scheme

The basic idea of the generic multiresolution fusion scheme is motivated by the fact that the human visual system is generally sensitive to local contrast changes, i.e. edges. Motivated from this insight, and in mind that both image pyramids and the wavelet transform result in a multiresolution edge representation, it is straightforward to build the fused image as a fused multiscale edge representation.

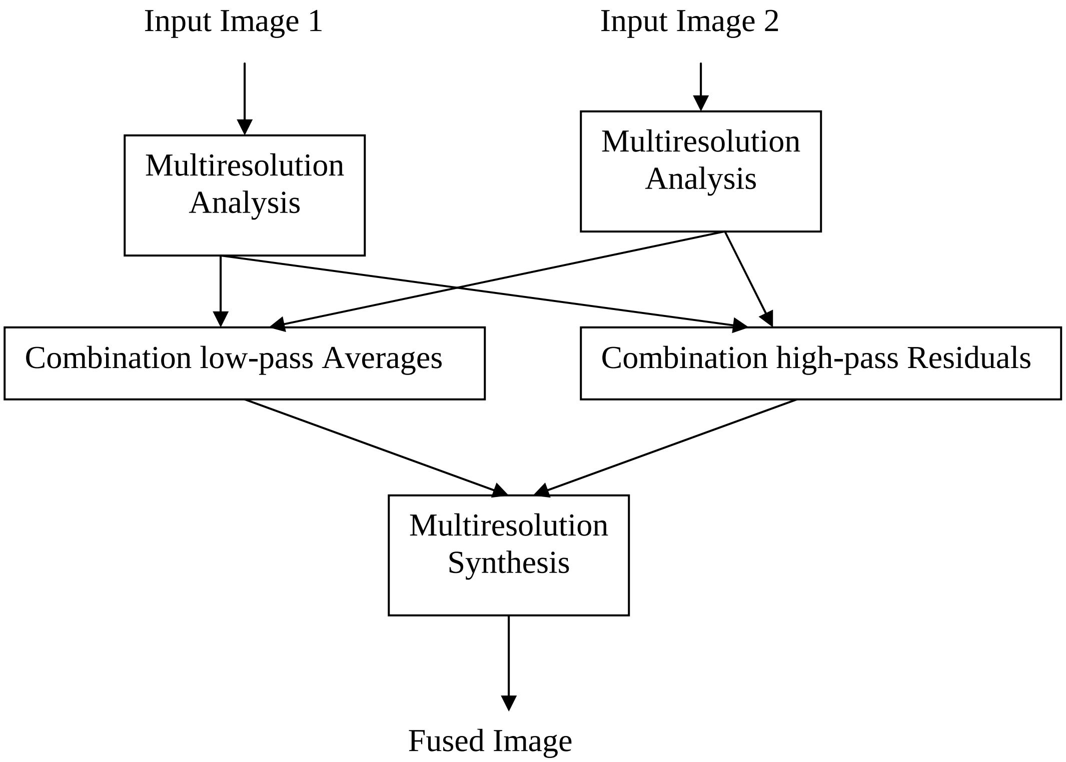


Figure 1.1: The generic multiresolution scheme

The fusion process is summarized in the Figure 1.1: In the first step the input images are decomposed into their multiscale edge representation, using either any image pyramid or any wavelet transform. The actual fusion process takes place in the synthesis, where the fused multiscale representation is built by a pixel-by-pixel selection of the coefficients with maximum magnitude. Finally the fused image is computed by an application of the appropriate reconstruction scheme.

1.5 MATLAB

This research has been completely implemented in MATLAB®. This high performance language for technical computer, integrates computation, visualization, and programming in an easy-to-use environment. One of the reasons of selecting MATLAB® in this research is because it fits perfectly in the necessities of an image processing research due to its inherent characteristics. MATLAB®'s basic data element is an array that does not require dimensioning. This is especially helpful to solve problems with matrix and vector formulations. And an image is nothing but a matrix or a set of matrices which defines the pixels value of the image, such as grey scale value in black and white images, and Red, Green and Blue or Hue, Saturation and Intensity values in color images.

In addition, MATLAB® includes an image processing toolbox which supports a wide range of image processing operations, including:

- ✓ Geometric operations
- ✓ Neighborhood and block operations
- ✓ Transforms
- ✓ Image analysis and enhancement
- ✓ Binary image operations
- ✓ Region of interest operations

However, this application has some limitations. Probably the most restricting is the computation time. A real time application should be implemented in some other time-efficient language such as C/C++ or similar.

1.6 Scope of Thesis

This work proposes a new approach that exploits the prevailing fundamental wavelet transform based image fusion approaches. Instead of using purely classical wavelet transform, a combination of wavelet transform, gradient and mathematical morphology has been used to eliminate the limitations associated with wavelet transform and to achieve better fusion results than classical wavelet transform.

1.7 Thesis Outline

Chapter 2 describes the theoretical and mathematical basis of wavelet transform. Then the next chapter explains the mathematical morphology; its advantages and reasons for its selection in our implementation. Chapter 4 provides the details of image gradients and activity level measurement. Chapter 5 will discuss the implementation methodology for the proposed approach. Chapter 6 presents the experimental results on standard test images against the recommended quality metrics and shows comparisons of the results achieved by the proposed scheme with results obtained from previous fusion techniques. Chapter 7 narrates conclusion and future work.

1.8 Summary

This chapter illustrates the need for image fusion and gives an overview of image fusion fundamentals and its different abstraction levels. Many fusion techniques i.e. linear superposition, non-linear methods, image pyramids and wavelet transform have been introduced. Among all these, wavelet transform fusion is most efficient. Moreover, this chapter also briefly discusses generic multiresolution image fusion scheme. Finally, scope of the thesis and structure of the thesis is outlined.

Chapter 2

THE WAVELET TRANSFORM

Although the Fourier transform has been the foundation of transform-based image processing since the late 1950s, but it is well suited only to the study of stationary data where all frequencies have an infinite coherence time. The Fourier transform brings only global information which is not sufficient to detect compact patterns in images. Gabor introduced a local Fourier transform, taking into account a sliding window, leading to a time frequency-analysis. This method is only applicable to situations where the coherence time is independent of the frequency. Morlet introduced the wavelet transform in order to have a coherence time proportional to the period.

2.1 Overview of Wavelet Transform

Images are generally connected regions of similar texture and gray level that combine to form objects. If the objects are small in size or low in contrast, they are examined at high resolutions; if they are large in size or high in contrast, a coarse view is all that is required. If both small and large objects (or low and high contrast objects) are present simultaneously, it can be advantageous to study them at several resolutions. Wavelet transform is concerned with the representation and analysis of images at more than one resolution.

Wavelet transform represents a windowing technique with variable-sized regions. Wavelet transform allows the use of long time intervals where we want more precise low-frequency information, and shorter regions where we want high-frequency information. Wavelet analysis is capable of revealing aspects of data (e.g., trends, breakdown points, discontinuities in higher derivatives, and self-similarity) that other image/signal processing techniques miss.

2.2 What is a Wavelet?

A wavelet is a waveform of effectively limited duration that has an average value of zero. Sinusoids, which are the basis of Fourier analysis, do not have limited duration; they extend from minus to plus infinity. Also, sinusoids are smooth and predictable, whereas wavelets tend to be irregular and asymmetric. This comparison of sine wave and a wavelet is shown in Figure 2.1.

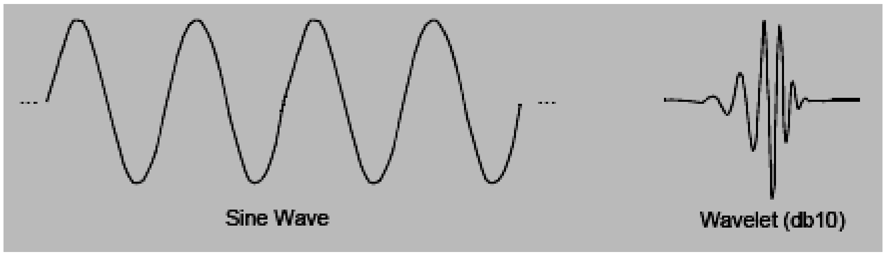


Figure 2.1: Comparison of a Sine Wave and a Wavelet [34]

Fourier analysis consists of breaking up a signal into sine waves of various frequencies. Similarly, wavelet analysis is the breaking up of a signal into shifted and scaled versions of the original (or *mother*) wavelet.

Just looking at pictures of wavelets and sine waves in Figure 2.1, it is evident that signals with sharp changes might be better analyzed with an irregular wavelet than with a smooth sinusoid. It also makes sense that local features can be described better with wavelets that have local extent.

2.2.1 Scaling

Scaling a wavelet simply means stretching (or compressing) it. To describe the effect of scaling, a new term, the *scale factor,* has been introduced, often denoted by the letter *a*. The smaller the scale factor, the more “compressed” the wavelet. The Figure 2.2 shows three versions of a wavelet with three different values of the scale factor. As the scale factor decreases, the wavelet is more compressed.

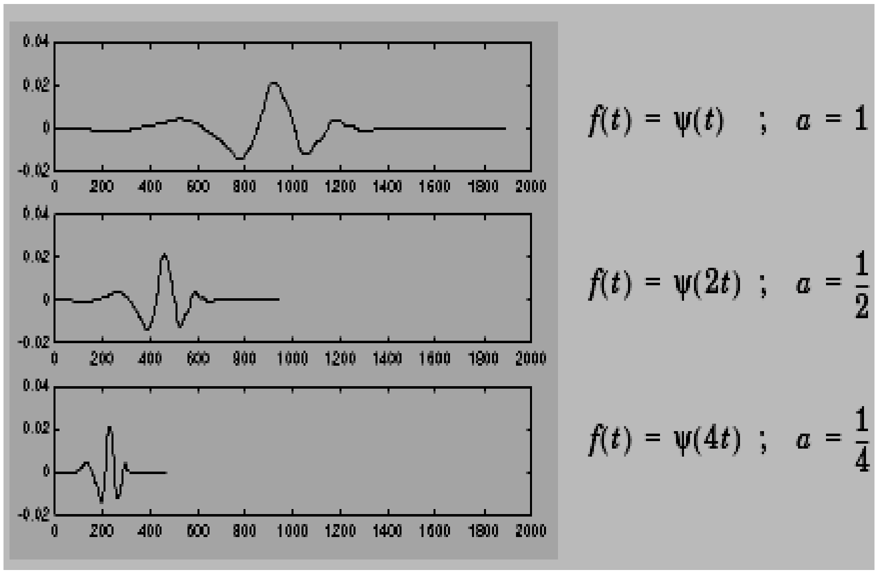


Figure 2.2: Effect of scale factor on a wavelet [34]

2.2.2 Shifting

Shifting a wavelet simply means delaying (or hastening) its onset. Mathematically, delaying a function *ψ (t)* by *k* is represented by *ψ (t-k)*. It is graphically shown in Figure 2.3 below.

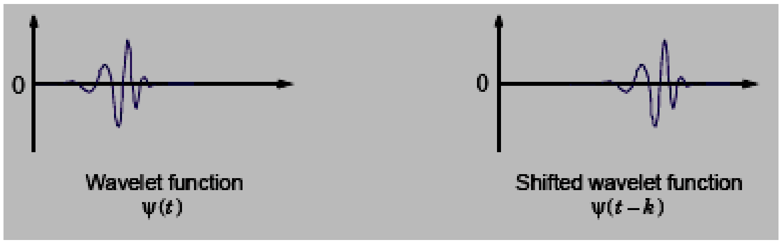


Figure 2.3: Effect of shifting a wavelet [34]

2.2.3 Scale and Frequency

As the higher scales correspond to the most “stretched” wavelets, the more stretched the wavelet, the longer is the portion of the signal with which it is being compared, and thus the coarser are the signal features being measured by the wavelet coefficients as depicted in the Figure 2.4.

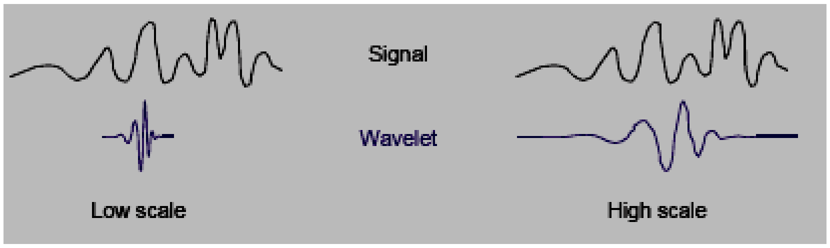


Figure 2.4: Relationship between scale and frequency of a wavelet [34]

Thus, there is a correspondence between wavelet scales and frequency as revealed by wavelet analysis:

- ✓ Low scale $\Rightarrow$ Compressed wavelet $\Rightarrow$ Rapidly changing details $\Rightarrow$ High frequency.
- ✓ High scale $\Rightarrow$ Stretched wavelet $\Rightarrow$ Slowly changing features $\Rightarrow$ Low frequency.

2.3 One-dimensional Discrete Wavelet Transform

The discrete wavelet transform (DWT) of a function $f(x)$ is given by the following pair of equations [33].

$$W_{\varphi}(j_0,k)=\frac{1}{\sqrt{M}}\sum_{x} f(x)\varphi_{j_0,k}(x) \tag{2.3-1}$$

$$W_{\psi}(j,k)=\frac{1}{\sqrt{M}}\sum_{x} f(x)\psi_{j,k}(x) \tag{2.3-2}$$

for $j \geq j_0$ and

$$f(x)=\frac{1}{\sqrt{M}}\sum_{k} W_{\varphi}(j_0,k)\varphi_{j_0,k}(x)+\frac{1}{\sqrt{M}}\sum_{j=j_0}^{\infty}\sum_{k} W_{\psi}(j,k)\psi_{j,k}(x) \tag{2.3-3}$$

Here $f(x)$, $\varphi_{j_0,k}(x)$ and $\psi_{j,k}(x)$ are functions of the discrete variable x = 0, 1, 2, …, M-1. Normally $j_0=0$ and M is a power of 2. The coefficients defined in Eqs. (2.3-2) and (2.3-3) are called approximation and detail coefficients respectively.

An efficient algorithm to implement DWT using filters was developed in 1988 by Mallat. This very practical filtering algorithm yields a *fast wavelet transform* (FWT) – a box into which a signal passes, and out of which wavelet coefficients quickly emerge.

For many signals, the low-frequency content is the most important part. It is what gives the signal its identity. The high-frequency content, on the other hand, imparts flavor or gradation. In wavelet analysis, the approximations are the high-scale, low-frequency components of the signal and the details are the low-scale, high-frequency components. The filtering process, at its most basic level, looks like as displayed in Figure 2.5:

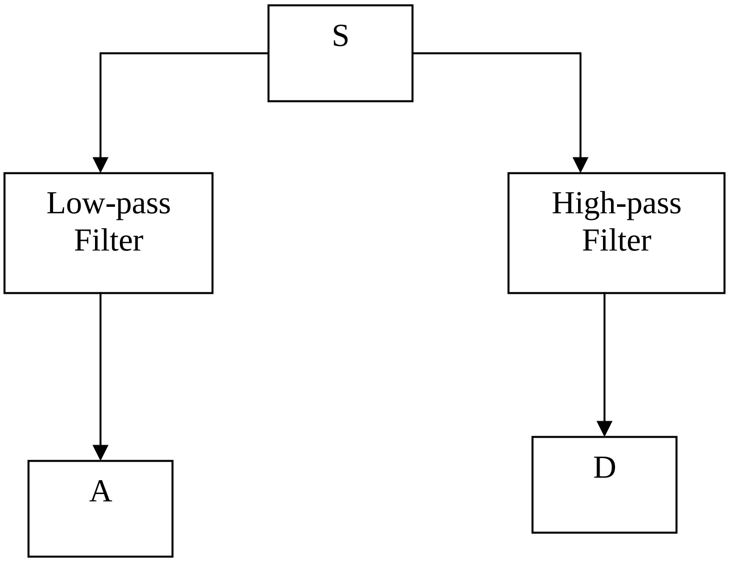


Figure 2.5: An FWT analysis bank

The original signal, S, passes through two complementary filters and emerges as two signals A(approximation) and D(detailed). If we actually perform this operation on a real digital signal, we wind up with twice as much data as we started with. Suppose, for instance, that the original signal S consists of 1000 samples of data. Then the resulting signals will each have 1000 samples, for a total of 2000. But there exists a more delicate way to perform the decomposition using wavelets. By looking carefully at the computation, we may keep only one point out of two in each of the two 2000-length samples to get the complete information.

The decomposition process can be iterated, with successive approximations being decomposed in turn, so that one signal is broken down into many lower resolution components. The Figure 2.6 shows decomposition of the signal S at the third level.

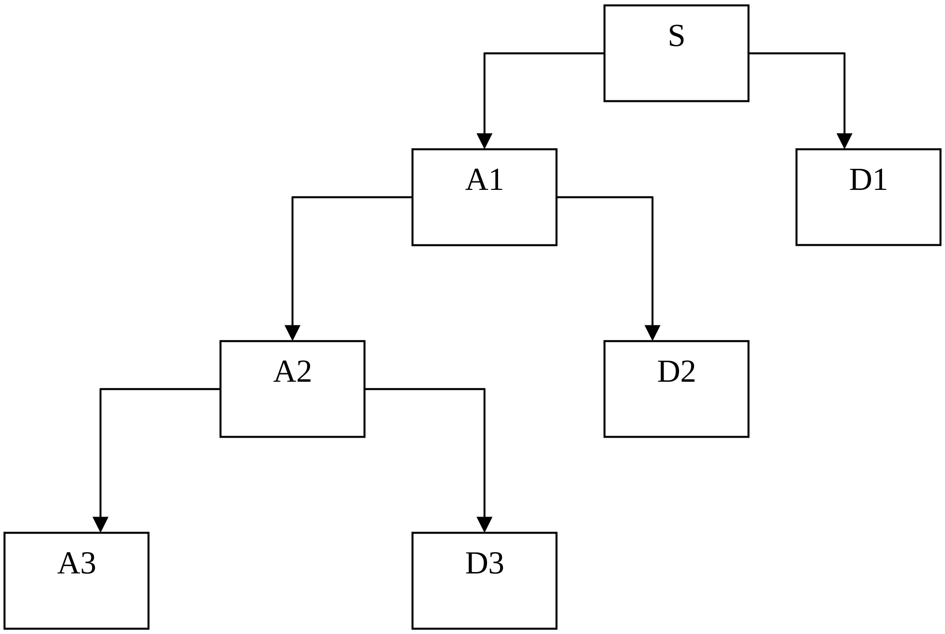


Figure 2.6: Multilevel wavelet-decomposition of a signal

2.4 Two-dimensional Discrete Wavelet Transform

In two dimensions, a two-dimensional scaling function, $\varphi(x, y)$ and three two dimensional wavelets $\psi^H(x, y)$, $\psi^V(x, y)$ and $\psi^D(x, y)$ are required [33]. Each is the product of a one-dimensional scaling function φ and corresponding wavelet ψ. Excluding products that produce one-dimensional results, like $\varphi(x)\varphi(x)$, the four remaining products produce the separable scaling function

$$\varphi(x, y) = \varphi(x)\varphi(y) \qquad (2.4\text{-}1)$$

and separable, “directionally sensitive” wavelets

$$\psi^H(x, y) = \psi(x)\varphi(y) \qquad (2.4\text{-}2)$$

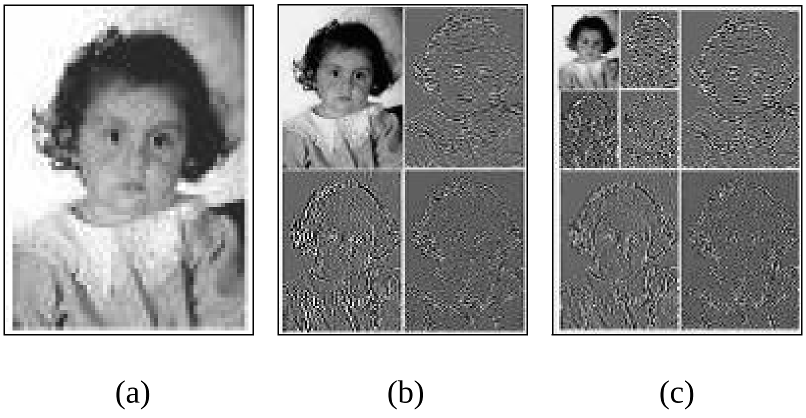
$$\psi^{V}(x,y) = \varphi(x)\psi(y) \tag{2.4-3}$$

$$\psi^{D}(x,y) = \psi(x)\psi(y) \tag{2.4-4}$$

These wavelets measure functional variations (i.e., intensity or gray-level variations for images) along different directions: ψ^{H} measures horizontal variations, ψ^{V} responds to vertical variations, and ψ^{D} corresponds to variations along diagonals.

Like the one-dimensional discrete wavelet transform, the two-dimensional discrete wavelet transform can be implemented using filters and downsamplers. With separable two-dimensional scaling and wavelet functions, we simply take the one-dimensional FWT of the rows of the two dimensional function $f(x,y)$, followed by the one-dimensional FWT of the resulting columns. Like its one-dimensional counterpart, the two-dimensional FWT filters the scale j + 1 approximation coefficients to construct the scale j approximation and detail coefficients. In the two-dimensional case, however, we get three detail coefficients – the horizontal, vertical and diagonal details.

The subimages, which are shown in Figure 2.7(b), are the inner products of the image in Figure 2.7(a) and the two-dimensional scaling and wavelet functions in Eqs. (2.4-1) through (2.4-4), followed by downsampling by two in each dimension. Two iterations of the filtering process produce the two-scale decomposition as shown in Figure 2.7(c).



(a) (b) (c)

Figure 2.7: The resulting decomposition of a two-dimensional FWT [35]

Chapter 3

Mathematical Morphology

3.1 Overview

It is a set-theoretic method of image analysis providing a quantitative description of geometrical structures [33]. The technique was originally developed by Matheron and Serra at the Ecole des Mines in Paris. It is called *morphology* since it aims at analyzing the shape and form of an object. It is *mathematical* in the sense that the analysis is based on the set theory, topology, lattice algebra and functions.

The primary application of morphology occurs in binary images, though it is also used on grey level images. It can also be useful on *range images*. (A range image is one where grey levels represent the distance from the sensor to the objects in the scene rather than the intensity of light reflected from them).

3.2 Morphological operations

The two basic morphological set transformations are *erosion* and *dilation.* These transformations involve the interaction between an image A (the object of interest) and a structuring set B, called the *structuring element*. Typically the structuring element B is a circular disc in the plane, but it can be any shape.

Let A and B be subsets in 2D plane. Then the *translation* of A by x, denoted A_x, is defined as

$$A_x = \{c : c = a + x,\ for\ a \in A\}, \tag{3.2-1}$$

the *reflection* of B, denoted $\hat{B}$, is defined as

$$\hat{B} = \{x : x = -b,\ for\ b \in B\}, \tag{3.2-2}$$

The dilation of the object A by the structuring element B, denoted $A \oplus B$, is defined as

$$A \oplus B = \{x : \hat{B}_x \cap A \neq \phi\}, \tag{3.2-3}$$

and the erosion of the object A by a structuring element B, denoted $A \,\Theta\, B$, is defined as

$$A \,\Theta\, B = \{x : B_x \subseteq A\}. \tag{3.2-4}$$

Erosion and dilation can be used in a variety of ways, in parallel and series, to give other transformations including thickening, thinning, skeletonization and many others. Two very important transformations are *opening* and *closing*. The opening of *A* by *B*, denoted $A \circ B$, is given by the erosion by *B*, followed by the dilation by *B*, that is

$$A \circ B = (A \,\Theta\, B) \oplus B \tag{3.2-5}$$

and the closing, denoted $A \bullet B$, is produced by the dilation of *A* by *B*, followed by the erosion by *B*:

$$A \bullet B = (A \oplus B) \,\Theta\, B \tag{3.2-6}$$

Opening and closing are the *duals* of each other, *i.e.* opening the foreground pixels with a particular structuring element is equivalent to closing the background pixels with the same element.

3.3 Applications of Morphological Operations

This section describes the applications of dilation, erosion, opening and closing morphological operations with reference to binary images. In the subsequent diagrams in this section, the structuring element is a 3×3 square, with the origin at its center, as shown in Figure 3.1.

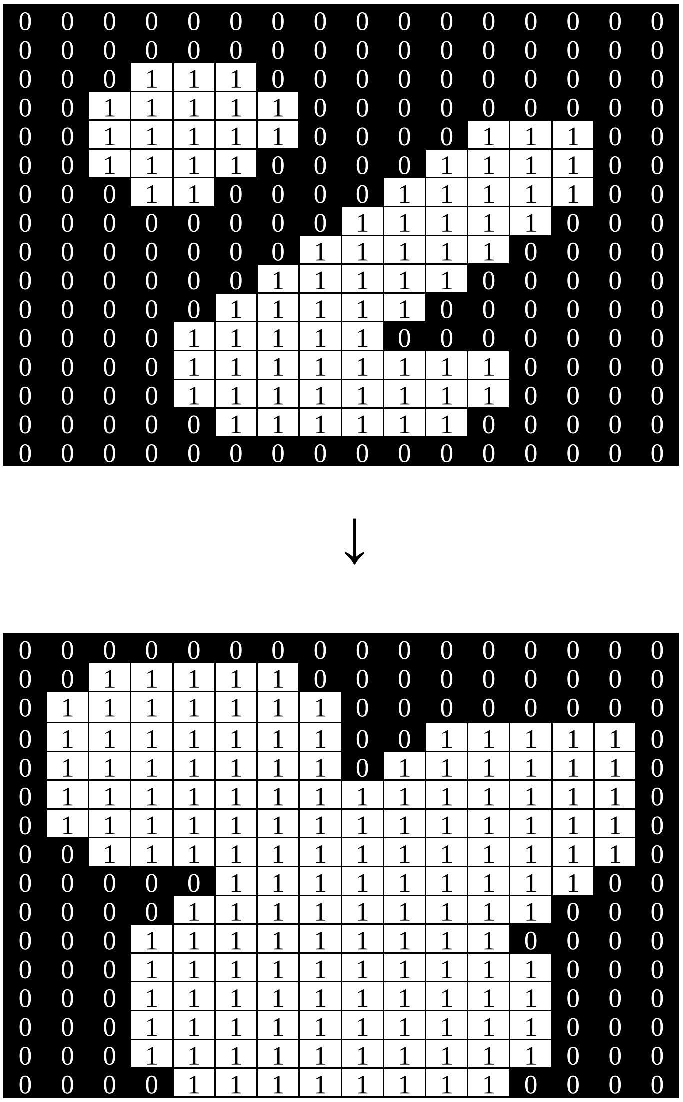
1	1	1
1	1	1
1	1	1

Figure 3.1: A 3×3 square structuring element

3.3.1 Dilation

To compute the dilation of a binary input image by the 3×3 square structuring element, we consider each of the pixels having value 0 in the input image in turn. For each of such pixels, we superimpose the structuring element on top of the input image so that the origin of the structuring element coincides with the 0-pixel position under consideration. If *at least one* pixel in the structuring element coincides with a pixel having value 1 in the image underneath, the 0-pixel under consideration is set to the value 1. Otherwise, it remains 0.

The effect of a dilation using the 3×3 square structuring element on a binary image is shown in Figure 3.2.



0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	0	0	0	0	1	1	1	0	0
0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0
0	0	0	1	1	0	0	0	0	1	1	1	1	1	0	0
0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0
0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0
0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0
0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0
0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0
0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0
0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0
0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

↓

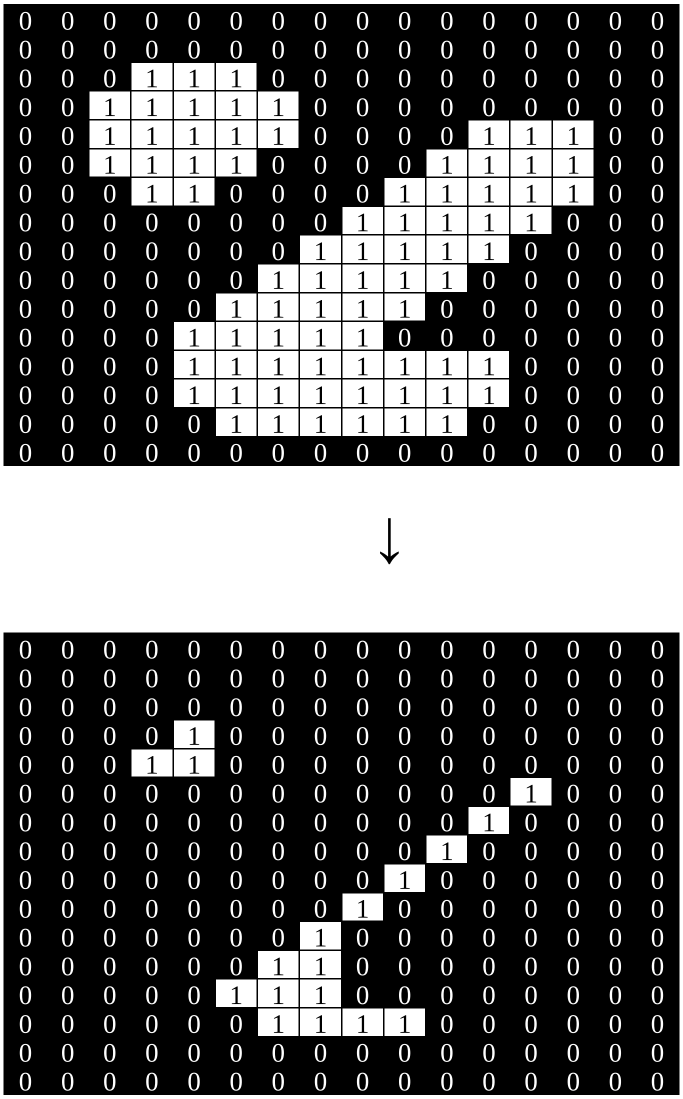
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0
0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
0	1	1	1	1	1	1	1	0	0	1	1	1	1	1	0
0	1	1	1	1	1	1	1	0	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0
0	0	0	0	0	1	1	1	1	1	1	1	1	1	0	0
0	0	0	0	1	1	1	1	1	1	1	1	1	0	0	0
0	0	0	1	1	1	1	1	1	1	1	1	0	0	0	0
0	0	0	1	1	1	1	1	1	1	1	1	1	0	0	0
0	0	0	1	1	1	1	1	1	1	1	1	1	0	0	0
0	0	0	1	1	1	1	1	1	1	1	1	1	0	0	0
0	0	0	1	1	1	1	1	1	1	1	1	1	0	0	0
0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0

Figure 3.2: Effect of dilation using a 3×3 square structuring element [36]

This example indicates that the effect of dilation on a binary image is to gradually enlarge the boundaries of regions of pixels having value 1. Thus areas of such pixels grow in size while holes within those regions become smaller.

3.3.2 Erosion

The effect of an erosion using the 3×3 square structuring element on a binary image is shown in Figure 3.3. To compute the erosion of a binary input image by this structuring element, we consider each of the pixels having value 1 in the input image in turn.



0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	0	0	0	0	1	1	1	0	0
0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0
0	0	0	1	1	0	0	0	0	1	1	1	1	1	0	0
0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0
0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0
0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0
0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0
0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0
0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0
0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0
0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

↓

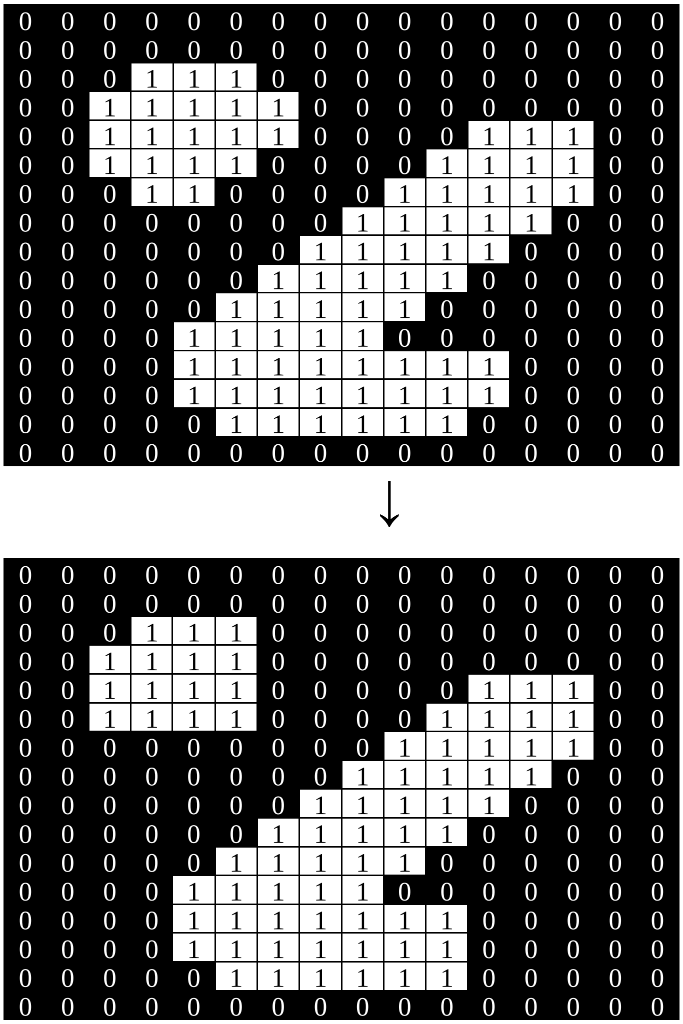
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 3.3: Effect of erosion using a 3×3 square structuring element [36]

For each of such pixels, we superimpose the structuring element on top of the input image so that the origin of the structuring element coincides with the 1-pixel position under consideration. If for every pixel in the structuring element the corresponding pixel in the image underneath has value 1, the 1-pixel under consideration is left as it is. Otherwise it is set to 0. This example illustrates that the effect of erosion on a binary image is to erode away the boundaries of regions of pixels having value 1. Thus areas of such pixels shrink in size, and the holes within those areas become larger.

3.3.3 Opening

The effect of an opening on a binary image using a 3×3 square structuring element is illustrated in Figure 3.4.



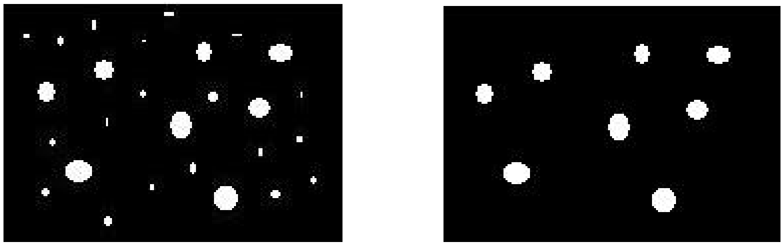
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	1	0	0	0	0	1	1	1	0	0
0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0
0	0	0	1	1	0	0	0	0	1	1	1	1	1	0	0
0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0
0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0
0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0
0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0
0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0
0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0
0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0
0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0
0	0	1	1	1	1	0	0	0	0	0	1	1	1	0	0
0	0	1	1	1	1	0	0	0	0	1	1	1	1	0	0
0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0
0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0
0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0
0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0
0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0
0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	0
0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0
0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0
0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 3.4: Effect of opening using a 3×3 square structuring element [36]

In the opening operation on the binary image, the structuring element is slides around *inside* each region of 1s, without changing its orientation. All the 1-pixels which can be covered by the structuring element with the structuring element being entirely within that region will be preserved. However, all 1-pixels which cannot be reached by the structuring element without parts of it moving out of the region of 1-pixels will be set to 0. After the opening has been carried out, the new boundaries of 1-pixels regions will all be such that the structuring element fits inside them, and so further openings with the same element have no effect.

Another application of opening is illustrated in Figure 3.5. Figure 3.5(a) is a binary image containing large holes and small holes of white color in a black background. If it is desired to remove the small holes while retaining the large holes, then we can simply perform an opening with a disk-shaped structuring element with a diameter larger than the smaller holes, but smaller than the large holes. The image in Figure 3.5(b) is the result of the opening.



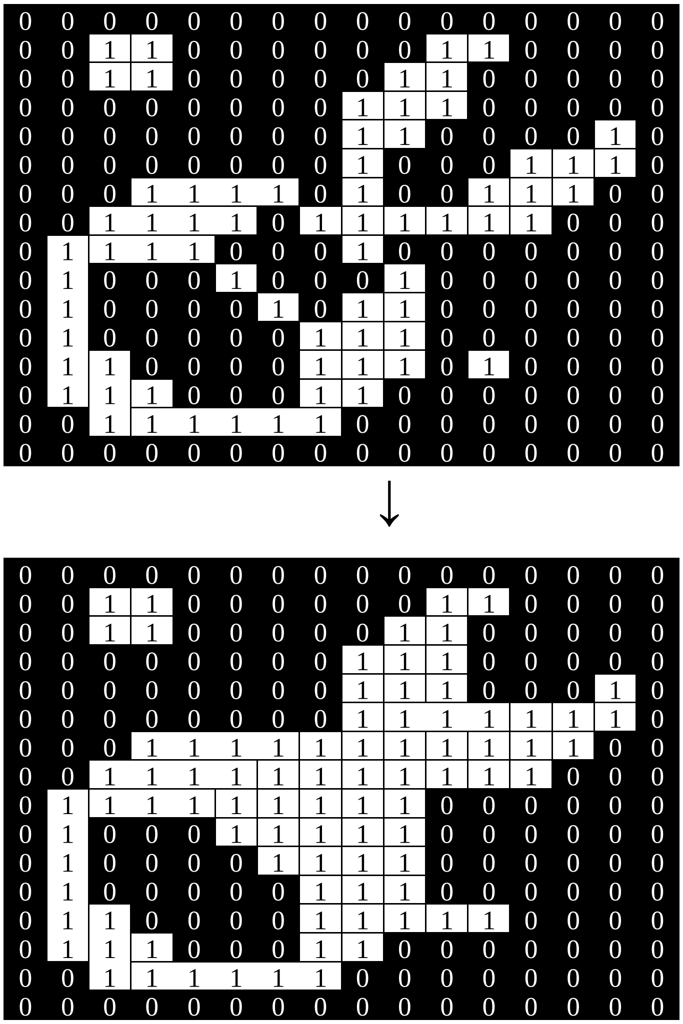
(a) (b)

Figure 3.5: Application of morphological opening operation

3.3.4 Closing

In closing operation on a binary image, the structuring element is slides around *outside* each region of pixels having value 1, without changing its orientation. For any 0-pixels boundary point, if the structuring element can be made to touch that point, without any part of the element being inside a 1-pixels region, that point remains 0. If this is not possible, the pixel is set to 1. After the closing has been carried out the 0-pixel regions will be such that the structuring element can be made to cover any point in the 0-pixel regions without any part of it also covering a 1-pixels point, and so, further closings will have no effect.

The effect of a closing on a binary image using the 3×3 square structuring element is illustrated in Figure 3.6.



0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0
0	0	1	1	0	0	0	0	0	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	0	0	0	0	1	0
0	0	0	0	0	0	0	0	1	0	0	0	1	1	1	0
0	0	0	1	1	1	1	0	1	0	0	1	1	1	0	0
0	0	1	1	1	1	0	1	1	1	1	1	1	0	0	0
0	1	1	1	1	0	0	0	1	0	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0
0	1	0	0	0	0	1	0	1	1	0	0	0	0	0	0
0	1	0	0	0	0	0	1	1	1	0	0	0	0	0	0
0	1	1	0	0	0	0	1	1	1	0	1	0	0	0	0
0	1	1	1	0	0	0	1	1	0	0	0	0	0	0	0
0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

↓

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0	0	0	1	1	0	0	0	0
0	0	1	1	0	0	0	0	0	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0	1	1	1	0	0	0	1	0
0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0
0	0	0	1	1	1	1	1	1	1	1	1	1	1	0	0
0	0	1	1	1	1	1	1	1	1	1	1	1	0	0	0
0	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0
0	1	0	0	0	1	1	1	1	1	0	0	0	0	0	0
0	1	0	0	0	0	1	1	1	1	0	0	0	0	0	0
0	1	0	0	0	0	0	1	1	1	0	0	0	0	0	0
0	1	1	0	0	0	0	1	1	1	1	1	0	0	0	0
0	1	1	1	0	0	0	1	1	0	0	0	0	0	0	0
0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Figure 3.6: Effect of closing using a 3×3 square structuring element [36]

Another application of closing is illustrated in Figure 3.7. The Figure 3.7(a) is an image containing large holes and small holes of black color in a white background. If it is desired to remove the small holes while retaining the large holes, we can simply perform a closing with a disk-shaped structuring element with a diameter larger than the smaller holes, but smaller than the large holes. The image in Figure 3.2(b) is the result of the closing.

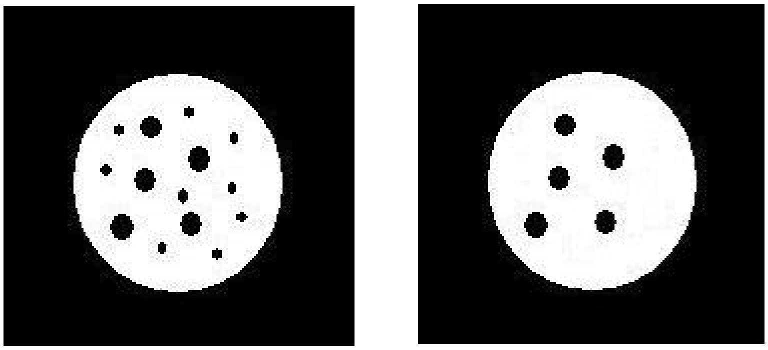

(a) (b)

Figure 3.7: Application of morphological closing operation

The morphological filter $(A \circ B) \bullet B$ can be used to eliminate *salt and pepper* noise. Salt and pepper noise is random, uniformly distributed small noisy elements often found corrupting real images. It will appear as black dots or small blobs on a white background, and white dots or small blobs on the black object. The background noise is eliminated at the erosion stage, under the assumption that all noise components are physically smaller than the structuring element *B*. Erosion on its own will increase the size of the noise components on the object. However, these are eliminated at the closing operation.

An important property of mathematical morphological operations is preservation of the main geometric structures of the object. Only features *smaller than* the structuring elements are affected by transformations. All other features at larger scales are not degraded.

Chapter 4

Gradient and Activity Level

The goal of this chapter is to deepen the understanding of the gradient and to illustrate how the gradient can be used for image fusion.

4.1 Introduction

The gradient of an image measures how it is changing. It provides two pieces of information. The magnitude of the gradient tells us how quickly the image is changing, while the direction of the gradient tells us the direction in which the image is changing most rapidly [33].

Because the gradient has a direction and a magnitude, it is natural to encode this information in a vector. The length of this vector provides the magnitude of the gradient, while its direction gives the gradient direction. Because the gradient may be different at every location, we represent it with a different vector at every image location.

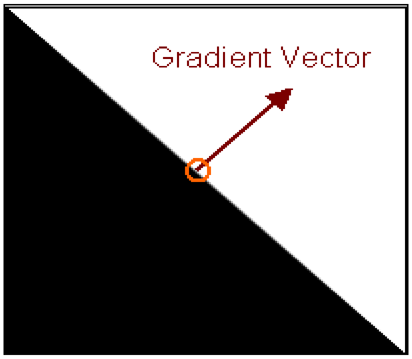


Figure 4.1: The gradient and an edge pixel

An edge in an image is a region where we have sharp contrast, that is, a rapid change in color intensity. A rapid change in a function's value gives a large magnitude of the gradient. Therefore, the gradient of an image will have a large magnitude at edges. If the color is roughly constant in an area of the image, the function is roughly constant in that area. A function that is roughly constant gives a small value of the gradient in that area.

In Figure 4.1, an ideal edge pixel and the corresponding gradient vector are shown. The circle indicates the location of the pixel. At the pixel, the intensity changes from 0 to 255 at the direction of the gradient.

4.2 Computing the Gradient

We form the gradient vector by combining the partial derivative of the image I in the *x* direction and the *y* direction. We can write this as:

$$\nabla I = \left[G_x, G_y\right] = \left[\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y}\right] \tag{4.2-1}$$

To explain this, let's review partial derivatives. When we take the partial derivative of *I* with respect to *x* we are determining how rapidly the image intensity changes as *x* changes. If *I* is a continuous two-dimensional function, we could write this as:

$$\frac{\partial I(x, y)}{\partial x} = \lim_{\Delta x \to 0} \frac{I(x + \Delta x, y) - I(x, y)}{\Delta x} \tag{4.2-2}$$

In the discrete case, we can only take differences at one pixel intervals. So we can take the difference between the pixel *I(x, y)* and the pixel before it, or the pixel after it. Or, we can treat the pixels before and after *I(x, y)* symmetrically, and compute:

$$\frac{\partial I(x, y)}{\partial x} \approx \frac{I(x+1, y) - I(x-1, y)}{2} \tag{4.2-3}$$

By similar reasoning, we can also compute:

$$\frac{\partial I(x, y)}{\partial y} \approx \frac{I(x, y+1) - I(x, y-1)}{2} \tag{4.2-4}$$

Combining Eqs. 4.2-3 and 4.2-4 with the Eq. 4.2-1 gives us a complete method for computing the image gradient.

4.3 Gradient Approximations

The magnitude of gradient vector given in Eq. (4.2-1) is given by

$$\nabla I = \left[G_x{}^2 + G_y{}^2 \right]^{1/2} \tag{4.3-1}$$

It is common practice to approximate the magnitude of the gradient by using absolute values instead of squares and square roots, that is:

$$\nabla I \approx \left| G_x \right| + \left| G_y \right| . \tag{4.3-2}$$

In the following subsections, digital approximations to the preceding equations are defined, and from there, the appropriate filter masks are formulated. Figure 4.2 denotes image points in a 3×3 region.

Z_1	Z_2	Z_3
Z_4	Z_5	Z_6
Z_7	Z_8	Z_9

Figure 4.2: A 3×3 region of an image

4.3.1 Roberts Cross-Gradient Operators

The approximations proposed by Roberts in the early development of digital image processing are given by:

$$G_x = (z_9 - z_5) \text{ and } G_y = (z_8 - z_6) \tag{4.3-3}$$

If we elect to use Eq. (4.3-1), then we compute the gradient as

$$\nabla I = \left[(z_9 - z_5)^2 + (z_8 - z_6)^2 \right]^{1/2} \tag{4.3-4}$$

If we use absolute values, substituting the quantities in Eq. (4.3-3) into Eq. (4.3-2) gives us the following approximation to the gradient given in Eq. (4.3-5).

$$\nabla I \approx |z_9 - z_5| + |z_8 - z_6| \qquad (4.3\text{-}5)$$

This equation can be implemented with the two masks shown in Figure 4.3. These masks are referred to as the *Roberts cross-gradient operators*. The primary disadvantage of the Roberts operator is its high sensitivity to noise, because very few pixels are used to approximate the gradient.

-1	0
0	1

0	-1
1	0

Figure 4.3: Roberts's gradient operators

4.3.2 Sobel Gradient Operators

The masks shown in Figure 4.4 are called *Sobel operators*, which are used to implement the gradient approximation given in Eq. (4.3-6).

$$\nabla I \approx |(z_7 + 2z_8 + z_9) - (z_1 + 2z_2 + z_3)| + |(z_3 + 2z_6 + z_9) - (z_1 + 2z_4 + z_7)| \qquad (4.3\text{-}6)$$

-1	-2	-1
0	0	0
1	2	1

-1	0	1
-2	0	2
-1	0	1

Figure 4.4: Sobel gradient operators

4.3.3 Compass Operators

Compass edge detection is an alternative approach to the differential gradient edge detection. The operation usually outputs two images, one estimating the local edge gradient magnitude and one estimating the edge orientation of the input image.

When using compass edge detection, the image is convolved with a set of (in general *8*) convolution kernels, each of which is sensitive to edges in a different orientation [33]. For each pixel the local edge gradient *magnitude* is estimated with the maximum response of all *8* kernels at this pixel location as given in Eq. (4.3-7).

$$|G| = \max(|G_i| : i = 1\ to\ n) \tag{4.3-7}$$

where G_i is the response of the kernel *i* at the particular pixel position and *n* is the number of convolution kernels. The local edge *orientation* is estimated with the orientation of the kernel that yields the maximum response. The maximum response |*G*| for each pixel is the value of the corresponding pixel in the output magnitude image. The values for the output orientation image lie between *1* and *n*, depending on which of the *n* kernels produced the maximum response.

Robinson

-1	1	1
-1	-2	1
-1	1	1

1	1	1
-1	-2	1
-1	-1	1

Sobel

-1	0	1
-2	0	2
-1	0	1

0	1	2
-1	0	1
-2	-1	0

Prewitt

1	1	1
0	0	0
-1	-1	-1

-1	0	1
-1	0	1
-1	0	1

Kirsch

-3	-3	5
-3	0	5
-3	-3	5

-3	5	5
-3	0	5
-3	-3	-3

Proposed

0	-3	0
0	6	0
0	-3	0

-3	0	0
0	6	0
0	0	-3

Figure 4.5: Common Variants of Compass Operators

Various kernels can be used for this operation; some of these are shown in Figure 4.5. Only two templates out of the set of *8* are shown. The whole set of *8* kernels is produced by taking one of the kernels and rotating its coefficients circularly. Each of the resulting kernels is sensitive to an edge orientation ranging from *0*° to *315*° in steps of *45*°.

The Roberts gradient operator, being two by two, responds best on sharp transitions in low-noise images and the other operators, being three by three, handle more gradual transition and noisier images better.

4.4 Activity Level Measurement

The activity level of a wavelet coefficient reflects the local energy in the space spanned by the term in the expansion corresponding to this coefficient. Higher the local energy value means clearer the pixel. There are three categories of methods for computing the activity level A(p) at position p: coefficient-based, window-based and region-based measures [35]. The coefficient-based activity measures consider each coefficient separately. The activity level is described by the absolute value or square of the corresponding coefficient in the wavelet decomposition representation. The window-based activity measures employ a small (typically 3×3 or 5×5) window centered at the current coefficient position. We also have several alternatives here. The regions used in region-based activity measurement are similar to windows with odd shapes. Region segmentation can be performed on the same band using the edge information and a labeling algorithm. The output is a labeled image in which each different value represents a different region. The proposed scheme computes activity level A(p) at position p by adding wavelet coefficient's absolute value |D(p)| to its local gradient G(p) as given in Eq. (4.4-1).

$$A(p) = \left|D(p)\right| + G(p) \qquad (4.4\text{-}1)$$

4.5 Summary

This chapter deals with the image gradient and activity level. The image gradient is important in boundary detection because images often change most quickly at the boundary between objects. The gradient can be used to find edges, which we define as places where the magnitude of the gradient is higher than the gradient magnitude at other locations that are in the direction of the gradient. Activity level is a measurement that distinguishes between the blur and clear pixels and it is computed using gradient.

Chapter 5

IMPLEMENTATION METHODOLOGY

This chapter deals with the design and implementation of proposed approach. MATLAB® has been chosen as a development tool because of the availability of required toolboxes and functions. The simulator is developed, compiled and tested in MATLAB®7.0.

5.1 Design of System

The block diagram of proposed scheme is shown in Figure 5.1 which actually consists of two phases. In phase-I, using wavelet transform and gradient, a binary decision-map is obtained which is then processed using morphological operations in phase-II to get the fused image.

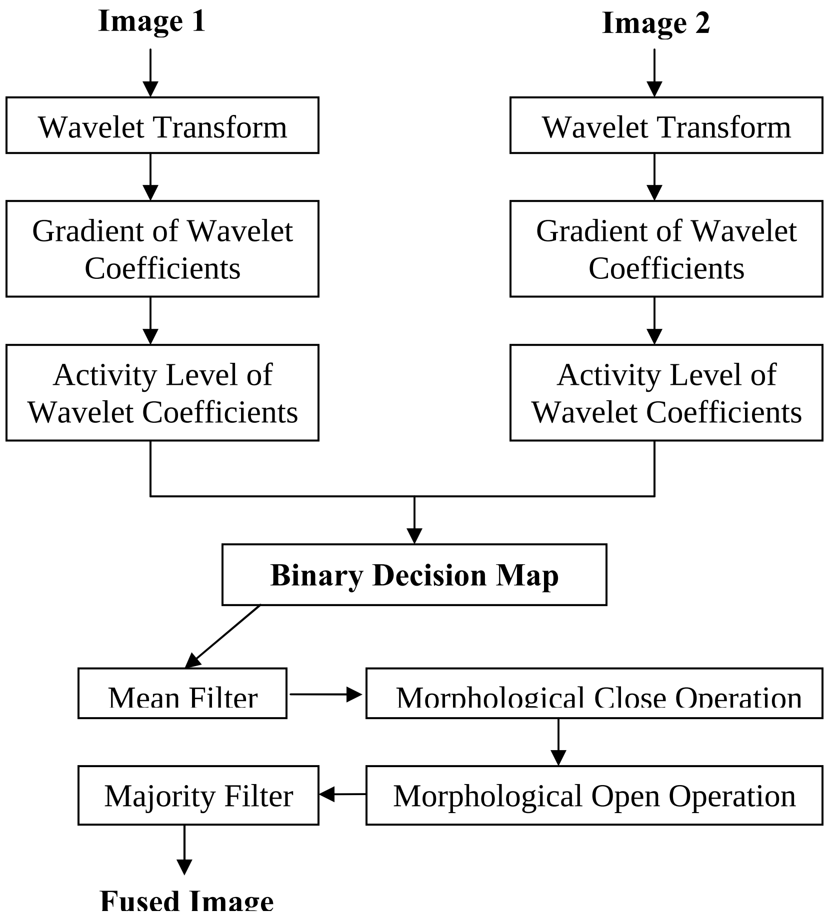


Figure 5.1: Block Diagram of proposed scheme

Block diagram of the proposed approach depicts the pseudo-code that is given below:

Pseudo-Code of Proposed Approach

The algorithm is follows:

1. Discrete Wavelet Transform is applied to obtain wavelet coefficients of the source images X and Y, both of size (M, N).
2. Detailed wavelet coefficients $xW_{detail}(m, n)$ and $yW_{detail}(m, n)$ at level 1, for source images X and Y respectively, are computed.

   where

   detail = horizontal, vertical and diagonal

   (m, n) = size of the wavelet coefficients matrices

3. Local gradient G of each detailed wavelet coefficient is computed as follows:

   $$G(\, W_{detail}(m, n)\,) = \max\{kernel_i * W_{detail}(m, n) \mid i = 1\text{-}4\}$$

   where, the four kernels are given below:

0	0	0
-3	6	-3
0	0	0

0	0	-3
0	6	0
-3	0	0

-3	0	0
0	6	0
0	0	-3

0	-3	0
0	6	0
0	-3	0

4. Image activity level of each wavelet coefficient is computed as follows:

   $$A(\, W_{detail}(m, n)\,) = G(\, W_{detail}(m, n)\,) + |\, W_{detail}(m, n)\,|$$

   where $A(\, W_{detail}(m, n)\,)$ reflects the activity level information of wavelet coefficient $W_{detail}(m, n)$.

5. Binary decision map BDM for the fused image is constructed as follows:

   $$BDM_{detail} = \begin{cases} 1 & if\ A(xW_{detail}) \geq A(yW_{detail}) \\ 0 & if\ A(xW_{detail}) < A(yW_{detail}) \end{cases}$$

   $$SBDM = \sum_{detail} BDM_{detail}$$

   $$BDM = \begin{cases} 1 & if\ \ SBDM \geq 2 \\ 0 & if\ \ SBDM < 2 \end{cases}$$

6. Apply following operations on BDM obtained in step 5 in the order given:
   i. Mean Filter

   ii. Mathematical Morphology Close Operation
   iii. Mathematical Morphology Open Operation
   iv. Majority Filter

7. Resize the binary decision map obtained in step 6 so that it equals the size of input images.
8. Finally the fused image *F(M, N)* is obtained as follows:

$$F(M,N)=\begin{cases} X(M,N) & if\ \ BDM(M,N)=1 \\ Y(M,N) & if\ \ BDM(M,N)=0 \end{cases}$$

5.1.1 Phase-I

Step 1- Acquiring Image

Acquire a greyscale standard test image of size 512 x 512 from the World Wide Web to be processed for the fusion process. Then using average filter mechanism, generate two multifocused images from it. Multifocused images can be of three different categories:

a) Left and right multifocused
b) Upper and lower multifocused
c) Inner and outer multifocused

The Figure 5.3 shows all these three types of multifocused images for the original Lena test image shown in Figure 5.2.

Figure 5.2: Original Lena test image of size 512 x 512

Figure 5.3: Multifocused images generated using average filter from Lena test image

Step 2- Application of Wavelet Transform

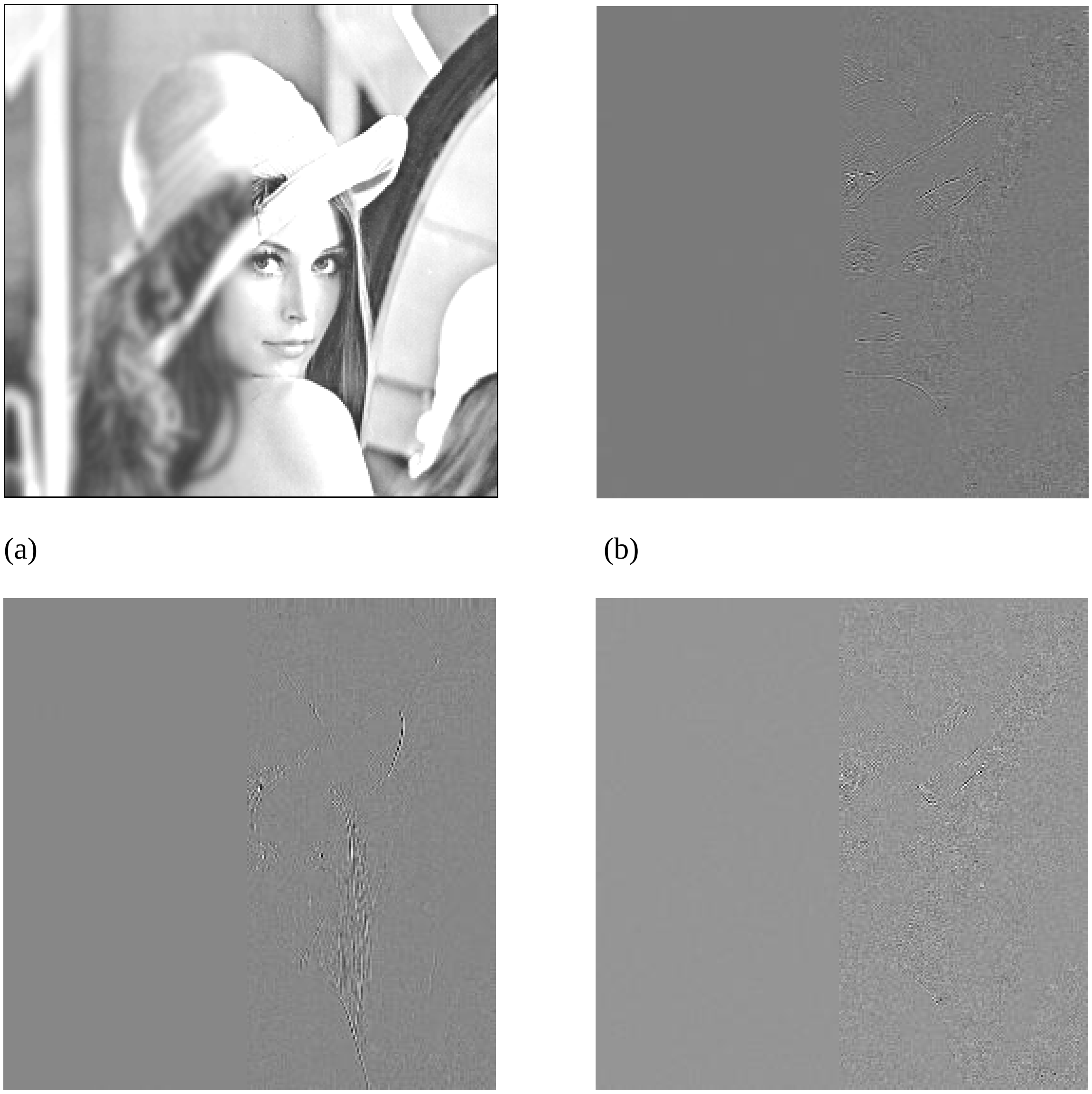
In transform phase, the first step is the application of fast discrete wavelet transform that transforms or decomposes the input image into an approximate image and three detailed images, that is, horizontal, vertical and diagonal details.

The Figure 5.4 and Figure 5.5 display wavelet coefficients (approximate, horizontal, vertical and diagonal) of left-focused and right-focused Lena images, respectively.

(a) (b)

(c) (d)

Figure 5.4: Wavelet coefficients of left-focused Lena image
(a) Approximate, (b) Horizontal, (c) Vertical and (d) Diagonal



(a) (b)

(c) (d)

Figure 5.5: Wavelet coefficients of right-focused Lena image (a) Approximate, (b) Horizontal, (c) Vertical and (d) Diagonal

Step 3- Computing Gradient of Wavelet Coefficients

The next step is calculation of local gradient of each of the wavelet coefficients, using kernels as given in step-3 of the pseudo-code. Local gradients are actually an intermediate result for the calculation of activity levels of the wavelet coefficients. Gradients are mainly used in images for edge detection and extraction. A rapid change in a function’s value gives a large magnitude of the gradient. Therefore, the gradient of an image will have large magnitude at edges because an edge in an image is a region where we have sharp contrast, that is, a rapid change in pixel intensity. On the other hand, if the pixel intensity is roughly constant in an area of the image, the function is roughly constant in the area. A function that is roughly constant gives a small value of the gradient in that area.

Step 4- Activity Level Measurement of Wavelet Coefficients

The activity level of a wavelet coefficient reflects the local energy in the space spanned by the term in the expansion corresponding to this coefficient. There are three categories of methods for computing the activity level $A_I(p)$ at position p: coefficient-based, window-based and region-based measures. In the proposed approach, the activity level of a wavelet coefficient is computed by adding the coefficient to its local gradient.

Step 5- Construction of Binary Decision Map

The last step in phase-I is the construction of a binary decision map, for the fused image, from activity levels of wavelet coefficients of the input images. For this, first we construct three decision maps one for each horizontal, vertical and diagonal detailed coefficients. Then a resulting decision map is computed from these three maps according to majority rule as given in step-5 of pseudo code in the previous section. This map is shown below in Figure 5.6.

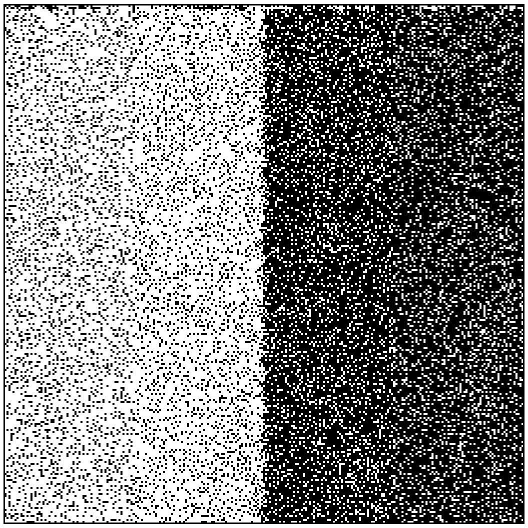


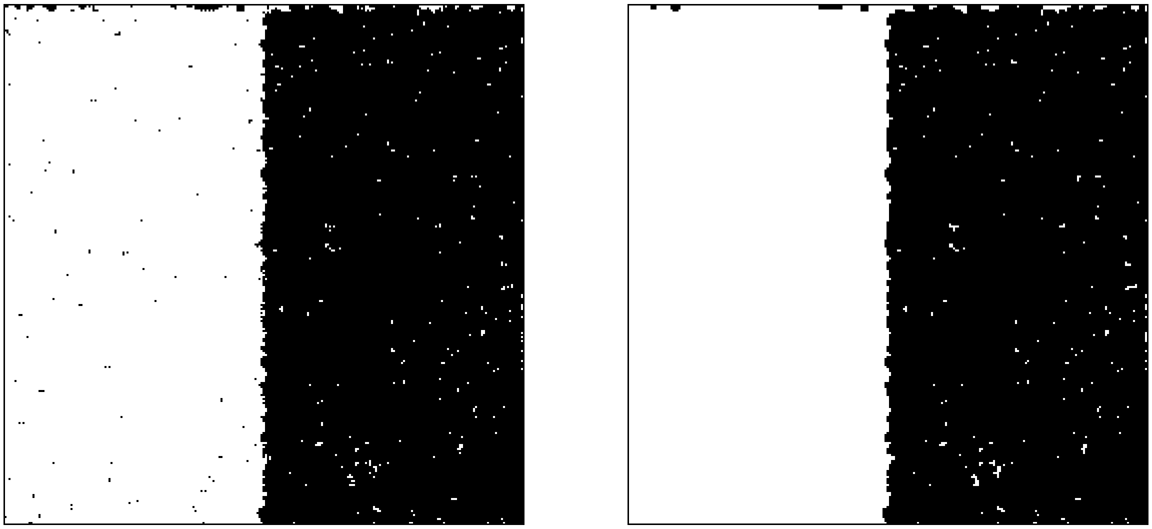
Figure 5.6: The binary decision map

5.1.2 Phase-II

The second phase deals with the application of morphological operations and filters to the binary decision map obtained as a result of phase-I.

Step 6- Binary Decision Map Operations

Theoretically the activity level measurement method is a proficient tool but the binary decision map obtained by comparing the activity level values is not good enough to properly classify the pixels in the input images into two groups of blur and clear pixels as can be seen in Figure 5.6. There are black dots in the white left-half and white dots in the black right-half of the decision map. Thus we have to improve the obtained binary decision map using some filters and morphological operations.



(a) (b)

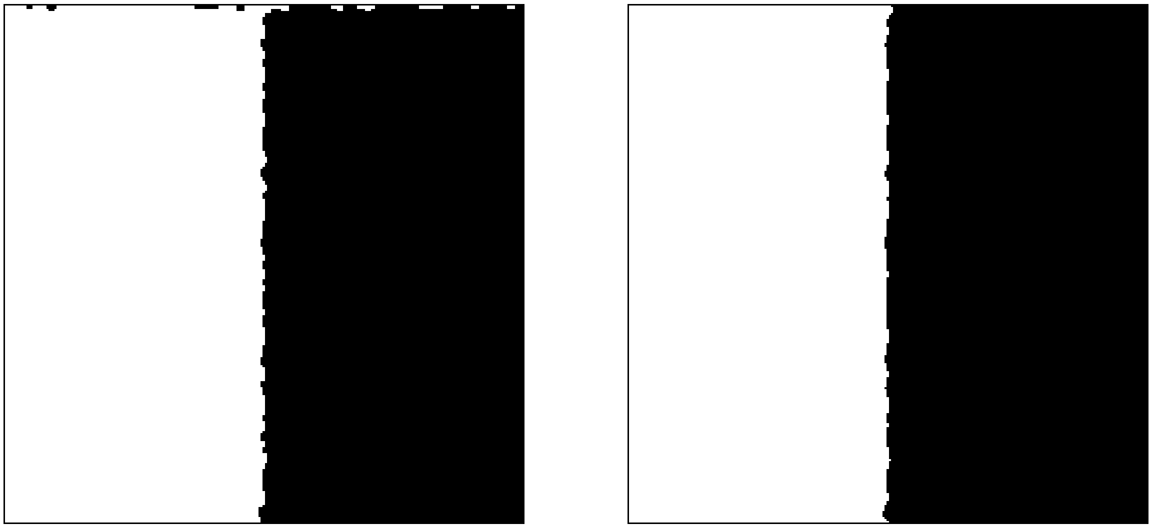

(c) (d)

Figure 5.7: The binary decision map operated by:
(a) mean filter, (b) morphological close, (c) morphological open and (d) majority filter

Firstly, the mean filter is applied on it to eliminate the noise effect. The Figure 5.7(a) is the result of mean filter operation. Then we use morphological close operation to fill the white left-half part, result shown in Figure 5.7(b); next, the morphological open operation is used to fill the black right-half part giving result as in Figure 5.7(c). The last operation on binary decision map is the majority filter that fills up any remaining holes as shown in Figure 5.7(d). This decision map is then resized so that it equals the size of the source images. Now, using this processed decision map, that clearly differentiates the two groups of pixels, the resulting fused image can be obtained.

Step 7- Construction of the fused image

In the last step, the resulting fused image *F(M, N)* against the input images X and Y, is constructed using the processed binary decision map *BDM(M, N)* according to the relation:

$$F(M,N)=\begin{cases} X(M,N) & if\ \ BDM(M,N)=1 \\ Y(M,N) & if\ \ BDM(M,N)=0 \end{cases}.$$

The fused image obtained against the left-focused and the right-focused Lena images is shown in Figure 5.8.

Figure 5.8: The resulting fused image

5.2 The Simulation

MATLAB® 7.0 has been used for the simulation and implementation of the proposed approach. A screen shot of the simulation is shown in Figure 5.9.

The foremost window of simulation allows the user to select an image and any defined multifocused option for the image fusion process. Three multifocusing options (left-right, upper-lower and inner-outer) are available here. The performance metrics i.e. Root Mean Square Error (RMSE), Peak Signal to Noise Ratio (PSNR) and Spatial Frequency (SF) at upper-right corner show the results obtained on the selected image with the selected multifocused option.

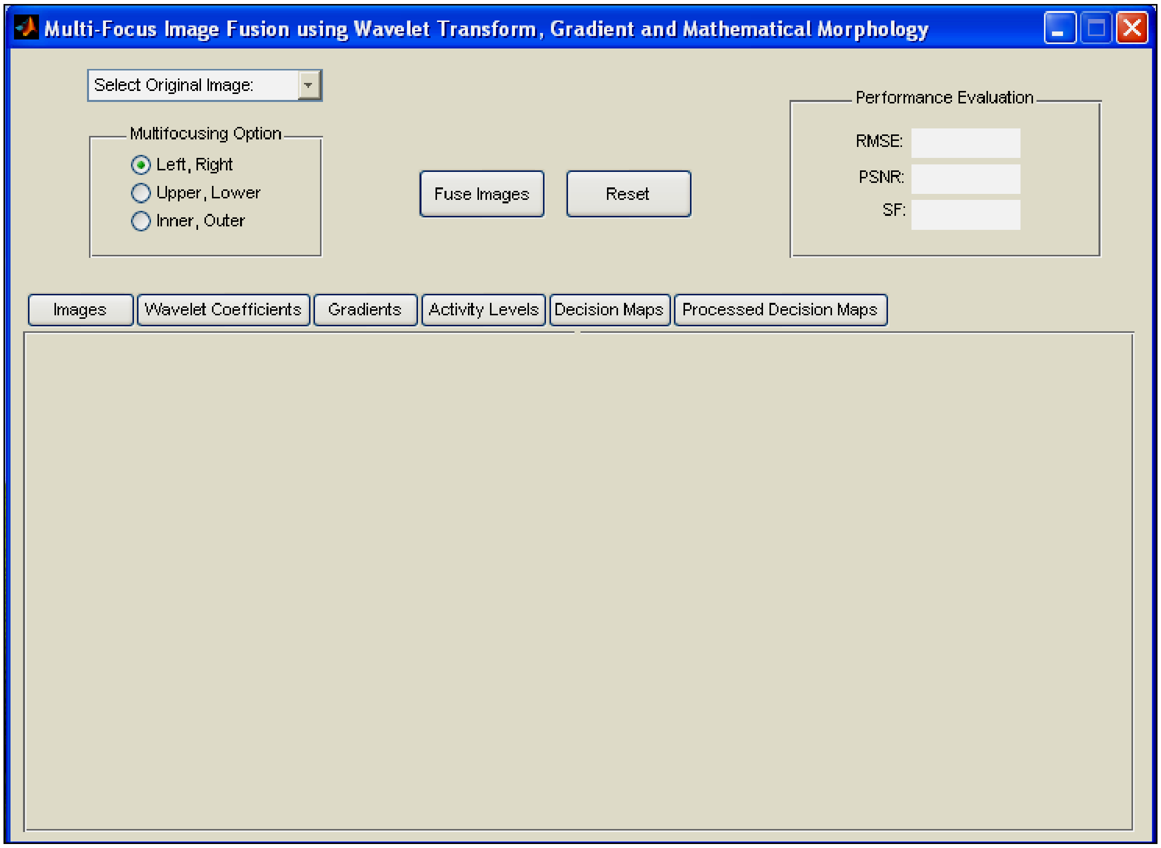


Figure 5.9: Screen shot of start window in MATLAB

Fineness of results after the fusion process on selected standard images has been observed. Different standard test images have been taken and fusion results conducted on these a number of times to get a high-quality image.

The Figure 5.10 shows the fusion results of upper-lower defocused Barbara test image.

Multi-Focus Image Fusion using Wavelet Transform, Gradient and Mathematical Morphology

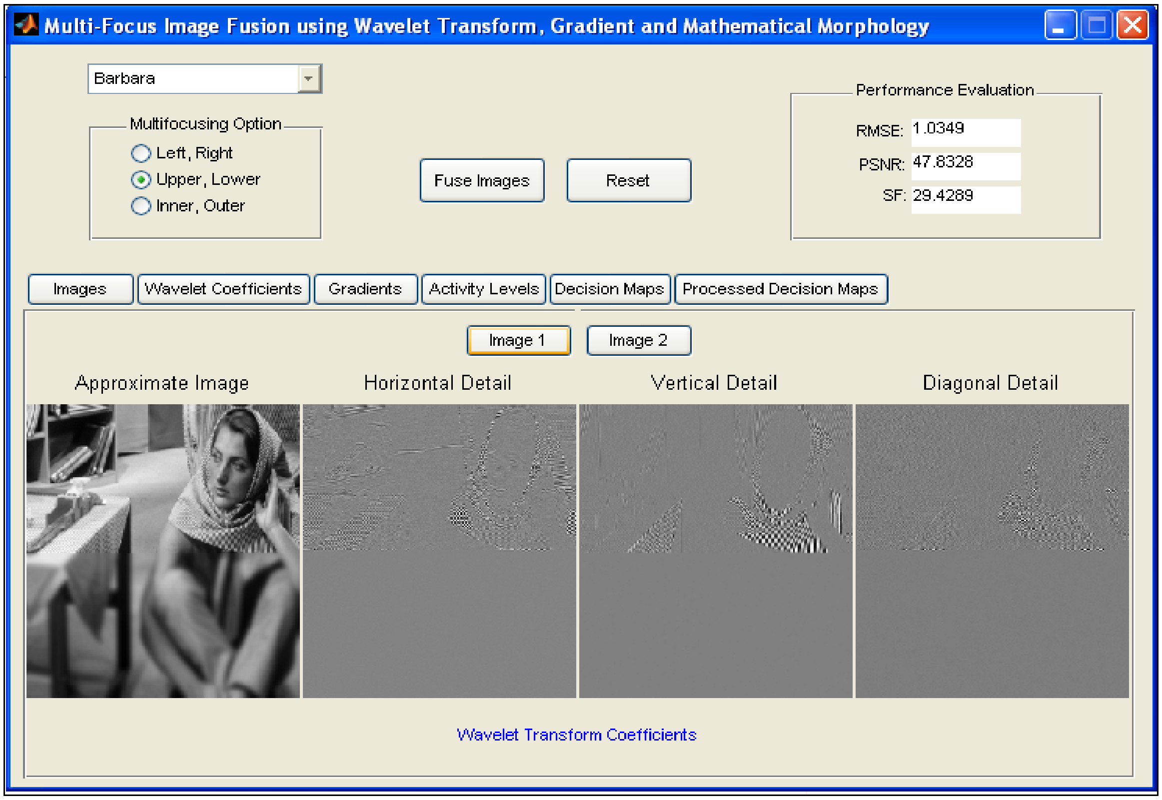
Barbara

Multifocusing Option
○ Left, Right
● Upper, Lower
○ Inner, Outer

Performance Evaluation
RMSE: 1.0349
PSNR: 47.8328
SF: 29.4289

Figure 5.10: Screen shot of simulation of proposed scheme in MATLAB

Figure 5.11 through Figure 5.13 display intermediate results of the fusion process. Figure 5.11 and Figure 5.12 depicts wavelets coefficients of the input images while Figure 5.13 shows the binary decision map at various stages during the process.

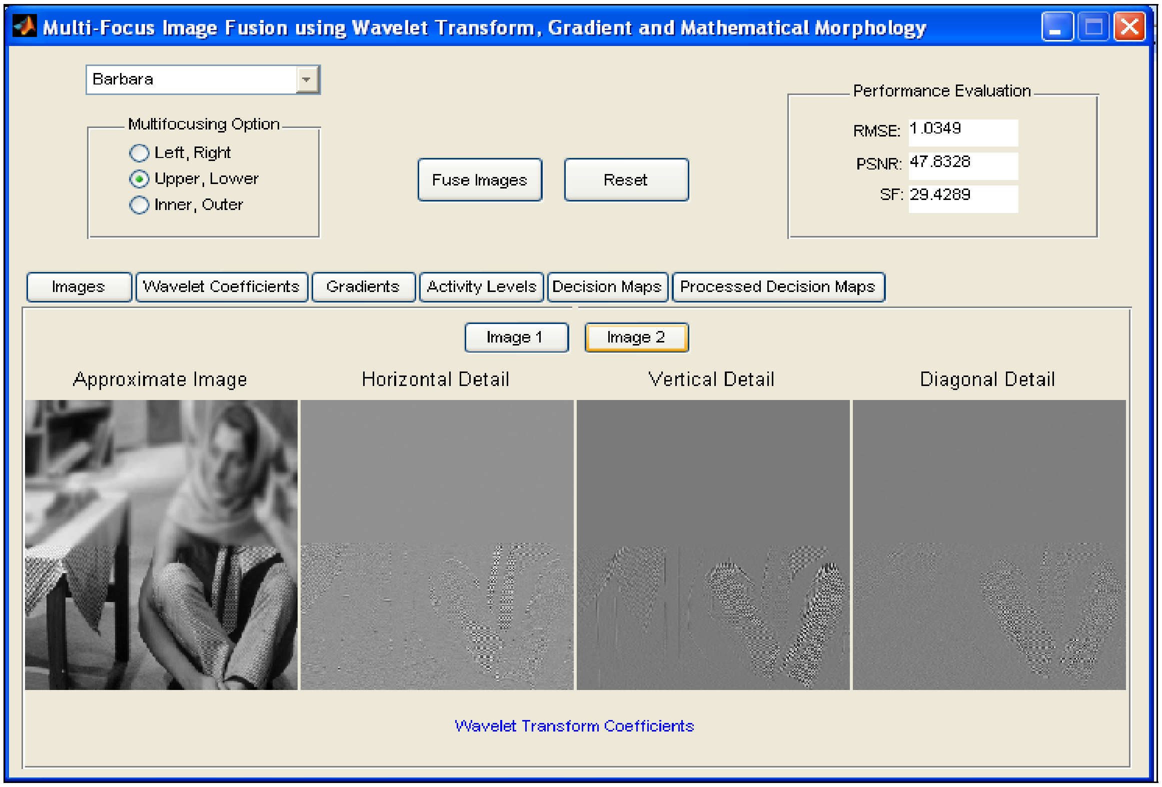


Barbara

Multifocusing Option
○ Left, Right
● Upper, Lower
○ Inner, Outer

Performance Evaluation
RMSE: 1.0349
PSNR: 47.8328
SF: 29.4289

Figure 5.11: Screen shot of wavelet coefficients of input image 1

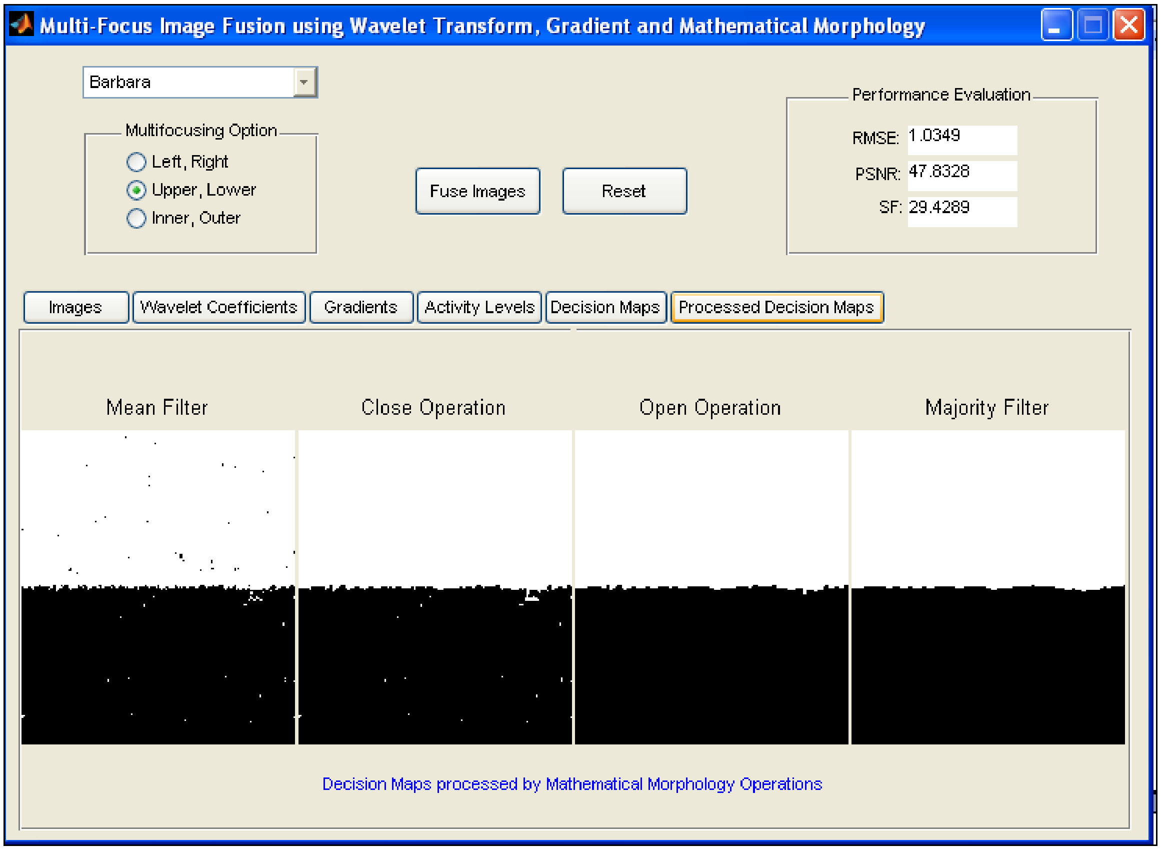


Barbara

Multifocusing Option
○ Left, Right
● Upper, Lower
○ Inner, Outer

Performance Evaluation
RMSE: 1.0349
PSNR: 47.8328
SF: 29.4289

Figure 5.12: Screen shot of wavelet coefficients of input image 2



Barbara

Multifocusing Option
○ Left, Right
● Upper, Lower
○ Inner, Outer

Performance Evaluation
RMSE: 1.0349
PSNR: 47.8328
SF: 29.4289

Figure 5.13: Screen shot of binary decision maps

5.3 Summary

This chapter deals with the entire design of the proposed approach along with its implementation in MATLAB. This chapter also includes graphical display of intermediate results and simulation screen shots of the main implementation.

Chapter 6

RESULTS AND DISCUSSION

Four grayscale standard test images (i.e. Lena, Barbara, Hill and Peppers) of size 512 x 512 have been taken from the World Wide Web for experiments and comparisons. MATLAB 7.0 has been used for the implementation of the proposed approach and results have been obtained on Pentium-IV, 3.20 GHz processor with a memory of 512 MB. Three different quality metrics i.e. Root Mean Square Error (RMSE), Peak Signal to Noise Ratio (PSNR) and Spatial Frequency (SF) are evaluated to compile fusion results. Results have been obtained for different multifocused regions (i.e. left-right, upper-lower and inner-outer) of the input images.

6.1 Performance Measures

Two types of metrics can be used to measure the performance of fusion system i.e. subjective and objective metrics. The following objective performance metrics have been analyzed to measure the quality of the reconstructed image.

- Peak Signal to Noise Ratio
- Root Mean Square Error
- Spatial Frequency

6.1.1 Peak Signal to Noise Ratio (PSNR)

This objective metric is used to measure the quality of the fused image. More value of PSNR indicate better quality image. For two $M \times N$ monochrome images I_R and I_F, where the first one is the original or reference image and the second is the fused image, it is defined as:

$$PSNR = 10.\log_{10}\left(\frac{\sum_{i=1}^{M}\sum_{j=1}^{N} 255^2}{\sum_{i=1}^{M}\sum_{j=1}^{N}\left[I_R(i,j) - I_F(i,j)\right]^2} \right).$$

6.1.2 Root Mean Square Error (RMSE)

When reference image is available, RMSE is the most valuable objective metric to measure and compare the quality of the fused images. Smaller value of RMSE indicates a better quality image. For a reference image I_R and the fused image I_F, both of size $M \times N$, RMSE is defined as:

$$RMSE = \left(\frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} \left[I_R(i, j) - I_F(i, j) \right]^2 \right)^{1/2} .$$

6.1.3 Spatial Frequency (SF)

Spatial frequency is used to assess the overall activity level of the fused image. Greater value of SF indicates better quality image. For the fused image I_F, it is defined as

$$SF = \sqrt{RF^2 + CF^2}$$

where RF and CF are the row frequency

$$RF = \left(\frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=2}^{N} \left[I_F(i, j) - I_F(i, j-1) \right]^2 \right)^{1/2}$$

and column frequency

$$CF = \left(\frac{1}{M \times N} \sum_{j=1}^{N} \sum_{i=2}^{M} \left[I_F(i, j) - I_F(i-1, j) \right]^2 \right)^{1/2} .$$

6.2 Experimental Results

Tables 6.1, 6.2, 6.3, and 6.4 show the comparisons of fused image and input images of Lena, Peppers, Barbara and Gold Hill respectively.

Table 6.1: Comparison with Input Images of Lena

Multifocused Regions	RMSE			PSNR			SF		
	Input Image 1	Input Image 2	Fused Image	Input Image 1	Input Image 2	Fused Image	Input Image 1	Input Image 2	Fused Image
Left-Right	8.0112	9.0023	0.2786	30.0568	29.0438	59.2299	9.0379	6.7497	10.9605
Upper-Lower	9.5939	7.2783	0.4894	28.4909	30.8903	54.3376	6.9894	8.8607	10.9593
Inner-Outer	9.1557	7.7442	0.7840	28.8970	30.3512	50.2444	7.5347	8.4123	10.9502

Table 6.2: Comparison with Input Images of Peppers

Multifocused Regions	RMSE			PSNR			SF		
	Input Image 1	Input Image 2	Fused Image	Input Image 1	Input Image 2	Fused Image	Input Image 1	Input Image 2	Fused Image
Left-Right	11.8229	11.7744	0.4248	26.6763	26.7120	55.5669	11.5686	11.7063	15.9937
Upper-Lower	10.7791	12.7270	0.5297	27.4791	26.0363	53.6504	12.1558	11.0967	15.9906
Inner-Outer	14.7310	7.9590	1.0267	24.7662	30.1136	47.9020	8.2633	14.2198	15.9733

Table 6.3: Comparison with Input Images of Barbara

Multifocused Regions	RMSE			PSNR			SF		
	Input Image 1	Input Image 2	Fused Image	Input Image 1	Input Image 2	Fused Image	Input Image 1	Input Image 2	Fused Image
Left-Right	17.2554	12.3477	0.6765	23.3923	26.2990	51.5258	12.0306	27.1216	29.4368
Upper-Lower	16.0392	13.8631	1.0349	24.0271	25.2936	47.8328	15.2878	25.4465	29.4289
Inner-Outer	18.7463	9.8957	1.4814	22.6725	28.2219	44.7174	14.9432	25.6358	29.3951

Table 6.4: Comparison with Input Images of Gold Hill

Multifocused Regions	RMSE			PSNR			SF		
	Input Image 1	Input Image 2	Fused Image	Input Image 1	Input Image 2	Fused Image	Input Image 1	Input Image 2	Fused Image
Left-Right	9.6669	11.6977	0.3906	28.4250	26.7688	56.2958	12.8481	10.3093	16.2533
Upper-Lower	12.1303	9.1488	0.5326	26.4534	28.9035	53.6027	10.0107	13.0777	16.2459
Inner-Outer	12.4965	8.5396	0.9601	26.1951	29.5020	48.4846	9.8123	13.2472	16.2363

It becomes obvious after observing results from these tables that fusion process improves the image quality as the RMSE value of fused image is consistently smaller whereas PSNR and SF values are larger than those of the input images.

Tables 6.5 to 6.8 show comparisons among the proposed approach and some existing methods such as choose maximum wavelet coefficients method (DWT-I), choose maximum wavelet coefficients gradient method (DWT-II), and the methods described in [21-26]. For DWT-based fusion schemes, the wavelet basis "db4" is used. The wavelet decomposition levels of DWT-I and DWT-II are six and five levels respectively. Consistency verification in a 3x3 window is only used for the DWT-II. The corresponding authors provide results of the schemes in [21-26].

Table 6.5: Image Fusion Methods' Comparisons for Lena Image

Multifocused Regions	Left and Right			Upper and Lower			Inner and Outer		
Method	RMSE	PSNR	SF	RMSE	PSNR	SF	RMSE	PSNR	SF
DWT-I	1.2983	45.8631	10.9503	1.4912	44.6603	10.9419	1.7996	43.0275	10.9311
DWT-II	1.0285	47.8866	10.9495	1.4414	44.9552	10.9340	1.8714	42.6874	10.9023
in [21]	1.3231	45.6991	10.9493	1.4927	44.6516	10.9462	1.8198	42.9303	10.9281
in [22]	3.6830	36.8069	10.5258	3.8159	36.4988	10.3979	3.9098	36.2877	10.4065
in [23]	5.4541	33.3963	9.9226	4.7532	34.5910	10.2802	5.0688	34.0327	10.1560
in [24]	1.1863	46.6466	10.9340	1.4524	44.8894	10.9149	1.4282	45.0348	10.9014
in [25]	1.1643	46.8096	10.8867	1.2434	46.2388	10.8619	1.3830	45.3141	10.8251
in [26]	0.5416	53.4578	10.9528	0.5572	53.2104	10.9577	1.0081	48.0611	10.9244
Proposed	**0.2763**	**59.3026**	**10.9605**	**0.4583**	**54.9077**	**10.9593**	**0.7667**	**50.4385**	**10.9502**

Table 6.6: Image Fusion Methods' Comparisons for Peppers Image

Multifocused Regions	Left and Right			Upper and Lower			Inner and Outer		
Method	RMSE	PSNR	SF	RMSE	PSNR	SF	RMSE	PSNR	SF
DWT-I	2.0240	42.0067	15.9618	2.0930	41.7153	15.9609	2.3561	40.6871	15.9408
DWT-II	1.6673	43.6906	15.9702	1.7773	43.1358	15.9624	2.1582	41.4490	15.9364
in [21]	2.1073	41.6563	15.9584	2.1442	41.5053	15.9576	2.3003	40.8952	15.9464
in [22]	5.7325	32.9640	15.2661	5.7087	33.0000	15.2792	5.6914	33.0265	15.1747
in [23]	7.1395	31.0574	14.4864	7.0215	31.2022	14.6844	6.3359	32.0947	15.2745
in [24]	5.4568	33.3921	14.8664	5.4703	33.3706	14.8659	5.5941	33.1762	14.8354
in [25]	1.8753	42.6694	15.8314	1.8649	42.7175	15.8313	1.9694	42.2442	15.8456
in [26]	1.3881	45.2826	16.0978	1.3163	45.7440	16.0916	1.3990	45.2144	15.9854
Proposed	**0.5098**	**53.9833**	**15.9937**	**0.5227**	**53.7665**	**15.9906**	**0.9984**	**48.1448**	**15.9733**

Table 6.7: Image Fusion Methods' Comparisons for Barbara Image

Multifocused Regions	Left and Right			Upper and Lower			Inner and Outer		
Method	RMSE	PSNR	SF	RMSE	PSNR	SF	RMSE	PSNR	SF
DWT-I	1.8389	42.8398	29.4333	2.0037	42.0940	29.4360	2.2532	41.0748	29.4051
DWT-II	1.3938	45.2469	29.4361	1.7583	43.2291	29.4259	2.3986	40.5317	29.3706
in [21]	1.7456	43.2919	29.4410	1.8919	42.5928	29.4394	2.1307	41.5603	29.4241
in [22]	5.2466	33.7333	28.3492	5.3947	33.4914	28.4519	5.6808	33.0427	28.3034
in [23]	6.8454	31.4228	28.8119	7.9284	30.1471	28.1963	7.1027	31.1023	28.5374
in [24]	2.3989	40.5306	29.3835	2.4298	40.4194	29.3763	2.5784	39.9039	29.3368
in [25]	1.7834	43.1058	29.3594	1.8687	42.7001	29.3477	1.9951	42.1315	29.2656
in [26]	1.3040	45.8255	29.4270	1.6118	43.9845	29.4125	1.8010	43.0206	29.3784
Proposed	**0.6731**	**51.5687**	**29.4368**	**1.2118**	**46.4625**	**29.4289**	**1.2793**	**45.9913**	**29.3951**

Table 6.8: Image Fusion Methods' Comparisons for Gold Hill Image

Multifocused Regions	Left and Right			Upper and Lower			Inner and Outer		
Method	RMSE	PSNR	SF	RMSE	PSNR	SF	RMSE	PSNR	SF
DWT-I	1.5195	44.4970	16.2341	1.4867	44.6863	16.2300	1.7770	43.1372	16.2287
DWT-II	1.2584	46.1344	16.2391	1.2877	45.9345	16.2296	1.8886	42.6082	16.2127
in [21]	1.5558	44.2917	16.2399	1.4736	44.7631	16.2374	1.7246	43.3971	16.2334
in [22]	4.3686	35.3239	15.7418	4.3668	35.3276	15.5824	4.5751	34.9228	15.5634
in [23]	6.2954	32.1504	15.2236	5.6175	33.1400	15.4538	5.9288	32.6714	15.4395
in [24]	1.1953	46.5814	16.2176	1.3641	45.4341	16.2008	1.9861	42.1709	16.1566
in [25]	1.2487	46.2017	16.1644	1.2513	46.1839	16.1562	1.3810	45.3271	16.1082
in [26]	0.6930	51.3158	16.2479	0.6881	51.3775	16.2375	1.1671	46.7883	16.2094
Proposed	**0.3899**	**56.3127**	**16.2533**	**0.5356**	**53.5536**	**16.2459**	**0.9535**	**48.5441**	**16.2363**

Figure 6.1 graphically demonstrates the comparisons among RMSE values of proposed scheme and the above methods for the left-right defocused regions of the source images. Similarly Figure 6.2 and Figure 6.3 present the comparisons among PSNR and SF values of these techniques.

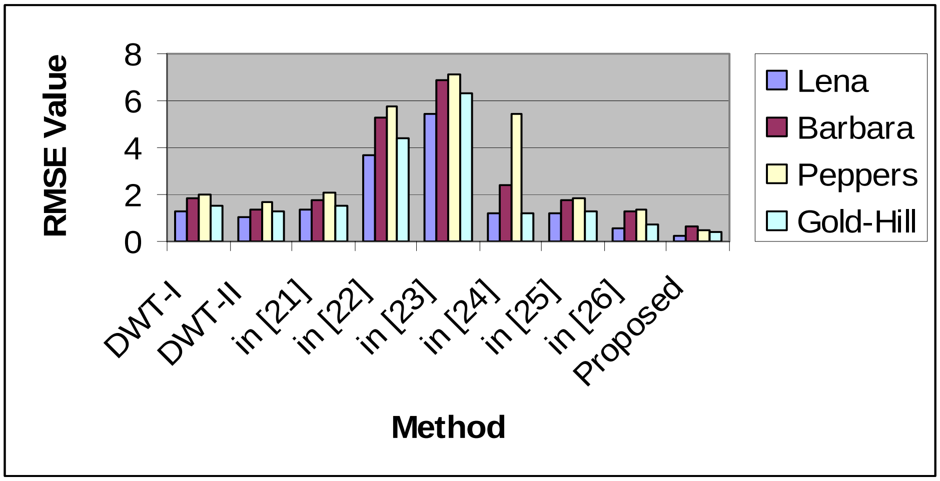


Figure 6.1: Comparisons of RMSE

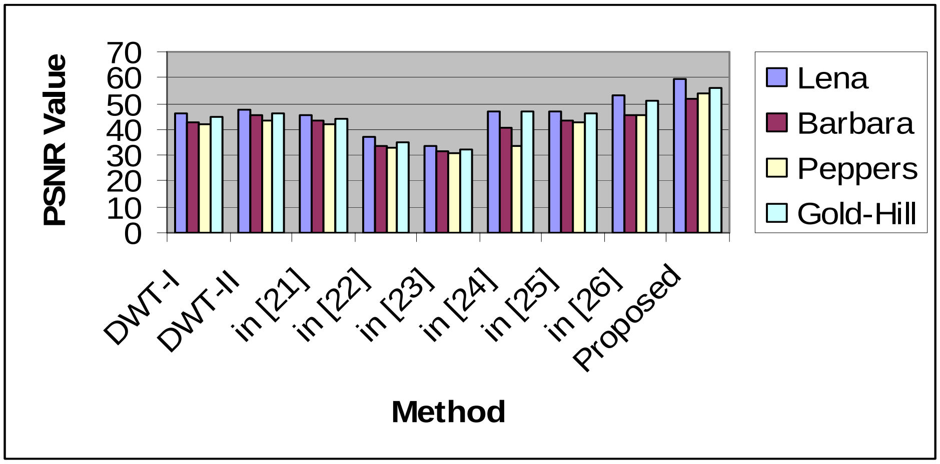


Figure 6.2: Comparisons of PSNR

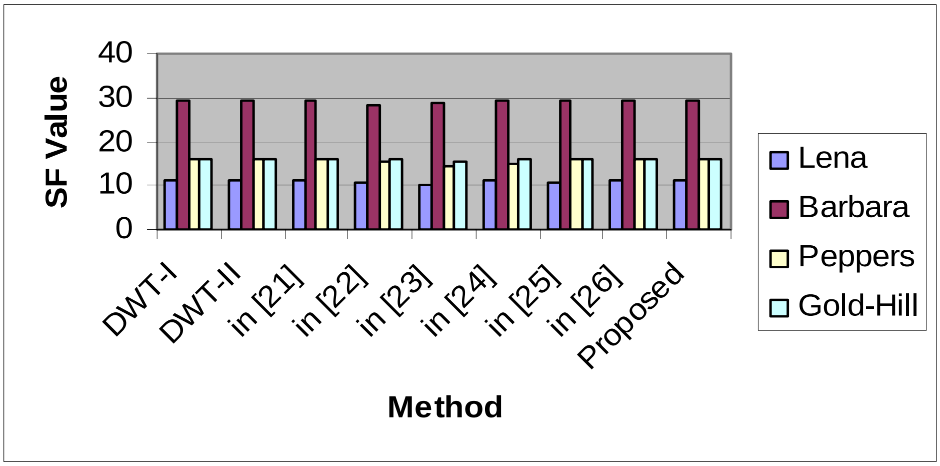


Figure 6.3: Comparisons of SF

Different defocused regions have slight variations in RMSE, PSNR and SF values. In case of left-right defocused regions, the result is better than that of upper-lower and inner-outer defocused regions. This is because different defocused regions have different clear and blur pixels due to which the edges are also different in each case. So the gradient operators give slightly different results.

The comparing results show that the proposed image fusion approach has the lower RMSE value and greater PSNR and SF values. It means that the proposed approach is comparatively better than the other methods discussed above.

6.3 Summary

In this chapter analysis of proposed scheme has been done for standard test images (i.e. Lena, Barbara, Hill and Peppers) and achieved results compared with previous techniques. The empirical results provide smaller RMSE and higher PSNR values than those provided by some of the previous approaches, which strengthen the idea of using an integrated scheme of wavelet transform, gradient and mathematical morphology.

Chapter 7

CONCLUSION AND FUTURE WORK

7.1 Conclusion

Image fusion has its applications in many fields such as computer vision, automatic object detection, robotics, remote sensing, military and law enforcement, medical imaging and manufacturing. Image fusion can be defined as the combination of a set of input images of the same scene, from different sources, with the aim to obtain new or more precise knowledge about the scene, which is more suitable for human and machine perception or further image-processing tasks such as segmentation, feature extraction and object recognition. The objective of image fusion is to generate a resultant fused image that describes the scene better than any single input image with respect to some relevant properties, by extracting all the useful information from the source images, and not introducing artifacts or inconsistencies, which will distract human observers or the following processing.

In the past, multifocus image fusion has been carried out using a variety of techniques. All fusion techniques are based on the principle of extracting the valuable information from source images to create fused image containing all objects 'in focus'. The three major techniques are based on region selection methods, multiscale decomposition methods and learning based methods. In the region selection methods, the input images are primarily separated in regions or into segments using a segmentation technique. From sets of such regions, one region per set is selected based on a sharpness criterion to form the final fused image. The value of the sharpness criterion increases and decreases as objects come into focus and go out of focus, or if the contrast changes in the scene.

In multiscale decomposition based methods, a variety of fusion schemes has been used. Most of these involve pixel exploitation or substitution at a detail level and the result is changes in the intensity values of the fused image. Learning based methods use training engines, which learn to distinguish between sharp and blurred areas on the basis of the given focused and unfocussed training data sets. We proposed a technique for image fusion that is actually an integration of multiscale wavelet transform, gradient and mathematical morphology schemes.

7.2 Future Work

This work presents a technique for image fusion that is actually an integration of multi-scale wavelet transform, gradient and mathematical morphology schemes. The proposed scheme is part of those early efforts that combine wavelets with other schemes to achieve better results. It observes the supremacy on purely classical wavelet based fusion techniques as empirical results achieved with proposed scheme for image fusion provide high PSNR and low RMSE than some of the previous approaches. The future work will be on integration of some other fusion techniques with wavelets to analyze the effect on RMSE and PSNR of the fused image.

REFERENCES

1. Dr. –Ing. Michael Heizmann, “Image Fusion Tutorial,” in IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems, Heidelberg, 2006.
2. A. Ben Hamza, Yun He, Hamid Karim and Alan Willsky, “A multiscale approach to pixel-level image fusion,” in Integrated Computer-Aided Engineering, vol. 12, Issue 2, 2005, pp. 135-146, ISSN: 1069-2509.
3. E. Lallier and M. Farooq, “A real time pixel-level based image fusion via adaptive weight averaging,” in Proceedings of the International Conference on Information Fusion, vol. 2, 2000.
4. J. Tian, Y. Zheng and L. Shen, “A Comparative study of Wavelet-based Pixel Level Image Fusion,” in Proceedings of the International Conference on Signal and Image Processing, 2004.
5. W. Rong, B. Fanliang, J. Hua and L. Lihua, “A Feature-Level Image Fusion Algorithm Based on Neural Networks,” in Proceedings of the International Conference on Bioinformatics and Biomedical Engineering, 2007, pp.821-824.
6. M. E. Ravinandan and Jharna Majumdar, “Feature Based Image Fusion,” in Proceedings of the International Conference on Cognition and Recognition.
7. F. Zhigang, F. Songling, L. Runshun and Z. Baojum, “Feature-level image fusion technique based on wavelet transform,” in Proceedings of the International Society for Optical Engineering, vol. 4919, 2002, pp. 289-292.

8. E. J. Bender and C. E. Reese, “Comparison of additive image fusion vs. feature-level image fusion techniques for enhanced night driving,” in Proceedings of the International Society for Optical Engineering, vol. 4796, 2003.
9. H. S. Alhichri and M. Kamel, “Virtual circles: a new set of features for fast image registration,” in Pattern Recognition Letters, vol.24, 2003, pp. 1181–1190.
10. P. R. Andersen and M. Nielsen, “Non-rigid registration by geometry constrained diffusion,” in Proceedings of the International Conference on Medical Image Analysis, vol. 5, 2001, pp. 81–88.
11. M. A. Audette, F. P. Ferrie and T. M. Peters, “An algorithmic overview of surface registration techniques for medical imaging,” in Proceedings of the International Conference on Medical image Analysis, vol. 4, 2000, pp. 201–217.
12. Y. Bentoutou, N. Taleb, M. Chikr El Mezouar and J. Jetto, “An invariant approach for image registration in digital subtraction angiography,” in Proceedings of the International Conference on Pattern Recognition, vol. 35, 2002, pp. 2853–2865.
13. S. C. Cain, M. M. Hayat and E. E. Armstrong, “Projection-based image registration in the presence of fixed-pattern noise,” in Proceedings of the International Conference on Image Processing, vol. 10, 2001, pp. 1860–1872.
14. L. Gang and L. Xue-Qin, “Pixel-Level image Fusion Based on Fuzzy Theory,” in International Conference on Machine Learning and Cybernetics, Vol. 3, 2007, pp. 1510 – 1514.

15. H. Chu, J. Li and W Zhu, “A Novel Support Vector Machine-Based Multifocus Image Fusion Algorithm,” in Proceedings of the International Conference on Communications, Circuits and Systems, Vol. 1, 2006, pp. 500 – 504.
16. L. Shutao, T. K. James, and W. Yaonan, “Multifocus image fusion using artificial neural networks,” in Proceedings of the International Conference on Machine Learning and Cybernetics, 2005, pp. 985 – 997.
17. Y. Xia and M. S. Kamel, “Novel Cooperative Neural Fusion Algorithms for Image Restoration and Image Fusion,” in IEEE Transactions on Image Processing, Vol. 16, Issue 2, Feb 2007, pp. 367 – 381.
18. P. Thevenaz and M. Unser, “A pyramid approach to sub-pixel image fusion based on mutual information,” in Proceedings of the International Conference on Image Processing, vol. 1, 1996, pp. 265-268.
19. Z. Liu, K. Tsukada, K. Hanasaki, Y.K. Ho and Y. P. Dai, “Image fusion by using steerable pyramid,” in Pattern Recognition Letters, vol. 22, Issue 9, July 2001, pp. 929-939.
20. Y. Song, M. Li, Q. Li and L. Sun, “A New Wavelet Based Multi-focus Image Fusion Scheme and Its Application on Optical Microscopy,” in IEEE International Conference on Robotics and Biomimetics, 2006, pp. 401-405.
21. C. Hua-Wen and L. Shu-Duo, “Image fusion based on addition of wavelet coefficients,” in International Conference on Wavelet Analysis and Pattern Recognition, Vol. 4, 2007, pp. 1585 – 1588.
22. Lei Tang and Zong-gui Zhao, “The Wavelet-based Contourlet Transform for Image Fusion,” in the Eighth ACIS International Conference on Software

Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, Vol. 2, 2007, pp. 59 – 64.

23. Muwei Jian, Junyu Dong and Yang Zhang, “Image Fusion Based on Wavelet Transform,” in the Eighth ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, Vol. 1, 2007, pp. 713 – 718.
24. Qu Xiaobo and Yan Jingwen, “Image Fusion Algorithm Based on Features Motivated Multi-Channel Pulse Coupled Neural Networks,” in the 2nd International Conference on Bioinformatics and Biomedical Engineering, 2008, pp. 2103 – 2106.
25. Xiao-Bo Qu, Guo-Fu Xie, Jing-Wen Yan, Zi-Qian Zhu and Ben-Gang Chen, “Image fusion algorithm based on neighbors and cousins information in nonsubsampled contourlet transform domain,” in International Conference on Wavelet Analysis and Pattern Recognition, Vol. 4, 2007, pp. 1797 – 1802.
26. Rodrigo Nava, Boris Escalante-Ramırez and Gabriel Cristobal “A novel multi-focus image fusion algorithm based on feature extraction and wavelets,” in *SPIE Photonics Europe, Optical and Digital Image Processing*, 7-11 April 2008, Strasbourge, France.
27. C. Myungjin, Y. K. Rae, N. R. Myeong and O. K. Hong, “Fusion of multispectral and panchromatic Satellite images using the curvelet transform,” in IEEE Letters of Geoscience and Remote Sensing, Vol. 2, Issue 2, 2005, pp. 136 – 140.
28. Z. W. Liao, S. X. Hu, and Y. Y. Tang, “Region-based Multifocus image fusion based on hough transform and wavelet domain hidden markov models,” in

Proceedings of the International Conference on Machine Learning and Cybernetics, vol. 9, 2005, pp. 5490-5495.

29. J. J. Lewis, R. J. O’Callaghan, S. G. Nikolov, D. R. Bull, and C. N. Canagarajah, “Region-based image fusion using complex wavelets,” in Proceedings of the Seventh International Conference on Information Fusion, vol. 1, 2004, pp. 555–562.
30. L. Yinghua, F. Xue, Z. Jingbo, W. Rujuan, Z. Kaiyuan and K. Jun, “A Multi-focus Image Fusion Based on Wavelet and Region Detection,” in EUROCON, 2007. The International Conference on Computer as a Tool, pp. 294 – 298.
31. Y. Liu, G. Baolong and N. Wei, “Multifocus Image Fusion Algorithm Based on Contourlet Decomposition and Region Statistics,” in Fourth International Conference on Image and Graphics, Aug. 2007, pp. 707-712.
32. http://www.metapix.de/methods.htm
33. Gonzalez and Woods, “Digital Image Processing,” 3rd Edition, ISBN: 9780131687288, Prentice Hall, 2008.
34. Michel Misiti, Yves Misiti, Georges Oppenheim and Jean-Michel Poggi, “Wavelet Toolbox 4 User’s Guide”.
35. P. Gonzalo, J. M Cruz, “A wavelet-based image fusion tutorial,” in Proceedings of the International Conference on Pattern Recognition, vol. 37, 2004, pp. 1855-1872.
36. http://homepages.inf.ed.ac.uk/rbf/HIPR2/matmorph.htm
37. http://homepages.inf.ed.ac.uk/rbf/HIPR2/prewitt.htm